**Ex.No:1**

**GUI Components, Fonts and Colors**

Date: 30/08/2022

**Aim:**

To implement hello world in android studio using GUI components, fonts

and colours.

**Software and Hardware Requirements:**

Software:

1. Android Studio
2. Java Hardware:
3. Preferably 8GB+ RAM Laptop/Desktop

**Algorithm:**

* 1. Open Android Studio and create a new project.
  2. Select Empty Activity.
  3. In the text attribute mention the text that has to be displayed.
  4. Background specifies the background colour

**Code:**

**MainActivity.java:**

package com.example.helloworld;

import androidx.appcompat.app.AppCompatActivity; import android.os.Bundle;

public class MainActivity extends AppCompatActivity {

@Override

protected void onCreate(Bundle savedInstanceState) { super.onCreate(savedInstanceState); setContentView(R.layout.activity\_main);

}

}

## activity\_main.xml:

<?xml version="1.0" encoding="utf-8"?>

<RelativeLayout xmlns:androi[d="http://schemas.android.com/apk/res/android"](http://schemas.android.com/apk/res/android) xmlns:app="[http://schemas.android.com/apk/res-auto"](http://schemas.android.com/apk/res-auto) xmlns:tools="<http://schemas.android.com/tools>" android:layout\_width="match\_parent" android:layout\_height="match\_parent"

android:background="#FFFD9F">

<!--

<androidx.constraintlayout.widget.ConstraintLayout xmlns:android="http:// schemas.android.com/apk/res/android"

xmlns:a[pp="http://schemas.android.com/apk/res-auto"](http://schemas.android.com/apk/res-auto) xmlns:tools="<http://schemas.android.com/tools>" android:layout\_width="match\_parent" android:layout\_height="match\_parent" android:background="#FFFD9F" tools:context=".MainActivity">

-->

<TextView android:layout\_width="196dp" android:layout\_height="64dp" android:layout\_alignParentStart="true" android:layout\_alignParentTop="true" android:layout\_alignParentEnd="true"

android:layout\_alignParentBottom="true" android:layout\_marginStart="101dp" android:layout\_marginTop="277dp" android:layout\_marginEnd="113dp" android:layout\_marginBottom="390dp" android:background="#E7E8E1" android:text="Hello World Ex1" android:textAlignment="center"

android:textColor="#E91E63" android:textSize="30sp" android:textStyle="bold"

app:layout\_constraintBottom\_toBottomOf="parent" app:layout\_constraintLeft\_toLeftOf="parent" app:layout\_constraintRight\_toRightOf="parent" app:layout\_constraintTop\_toTopOf="parent" android:layout\_alignParentLeft="true" android:layout\_alignParentRight="true" android:layout\_marginLeft="101dp" android:layout\_marginRight="113dp" android:gravity="center\_horizontal" />

<TextView android:id="@+id/textView" android:layout\_width="117dp" android:layout\_height="41dp"

android:layout\_alignParentStart="true" android:layout\_alignParentTop="true" android:layout\_marginStart="24dp" android:layout\_marginTop="28dp" android:fontFamily="casual" android:text="Top left" android:textColor="#ED7A15" android:textSize="30sp" android:textStyle="bold" app:layout\_constraintStart\_toStartOf="parent" app:layout\_constraintTop\_toTopOf="parent" android:layout\_alignParentLeft="true" android:layout\_marginLeft="24dp" />

<TextView android:id="@+id/textView2" android:layout\_width="155dp" android:layout\_height="41dp" android:layout\_alignParentTop="true" android:layout\_alignParentEnd="true" android:layout\_marginTop="25dp"

android:layout\_marginEnd="20dp" android:fontFamily="serif" android:text="Hi" android:textColor="#044375" android:textSize="30sp" android:textStyle="bold|italic" app:layout\_constraintEnd\_toEndOf="parent" app:layout\_constraintTop\_toTopOf="parent" android:layout\_alignParentRight="true" android:layout\_marginRight="20dp" />

<TextView android:id="@+id/textView3" android:layout\_width="159dp" android:layout\_height="51dp" android:layout\_alignParentStart="true"

android:layout\_alignParentBottom="true" android:layout\_marginStart="28dp" android:layout\_marginBottom="23dp" android:fontFamily="cursive" android:text="Bottom Left" android:textColor="#E56FFA" android:textSize="30sp" android:textStyle="bold"

app:layout\_constraintBottom\_toBottomOf="parent" app:layout\_constraintStart\_toStartOf="parent" android:layout\_alignParentLeft="true" android:layout\_marginLeft="28dp" />

<TextView android:id="@+id/textView4" android:layout\_width="162dp" android:layout\_height="33dp" android:layout\_alignParentEnd="true"

android:layout\_alignParentBottom="true" android:layout\_marginEnd="27dp" android:layout\_marginBottom="27dp" android:fontFamily="sans-serif-smallcaps"

android:text="Thank You" android:textColor="#EC2020" android:textSize="24sp" android:textStyle="bold"

app:layout\_constraintBottom\_toBottomOf="parent" app:layout\_constraintEnd\_toEndOf="parent" android:layout\_alignParentRight="true" android:layout\_marginRight="27dp" />

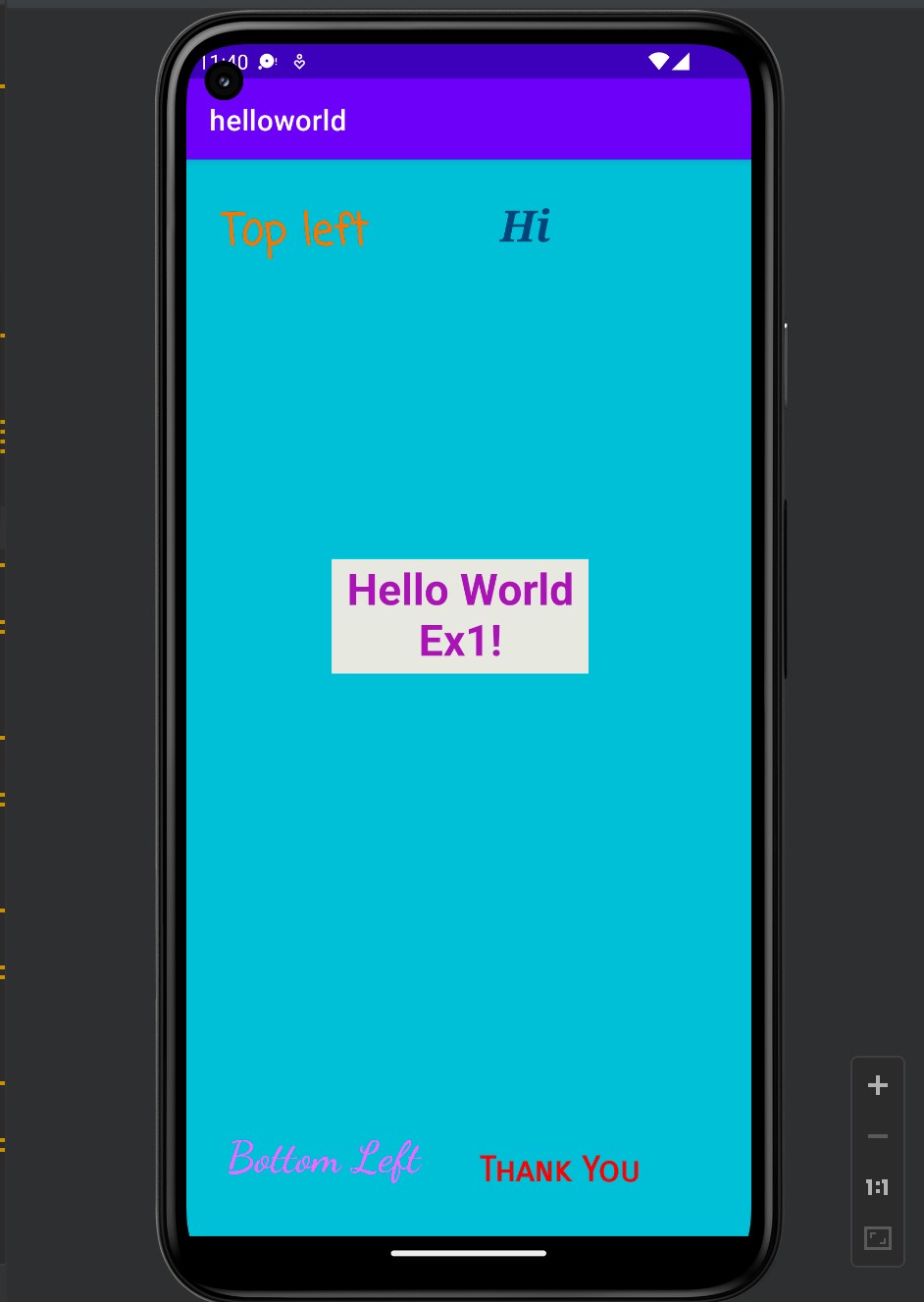
</RelativeLayout>

<!--

</androidx.constraintlayout.widget.ConstraintLayout>

-->

## Sample I/O:



**Result:**

Thus, hello world was successfully implemented in Android Studio using GUI Components, fonts and colours.

**Layout Managers and Event Listeners**

**Ex.No: 2**

Date: 06/09/2022

**Aim:**

To develop a Simple Android Application that uses Layout Managers and Event Listeners.

**Software and Hardware Requirements:**

Software:

1. Android Studio
2. Java Hardware:

1. Preferably 8GB+ RAM Laptop/Desktop

**Algorithm:**

1. Open Android Studio and create a new project.
2. Select Empty Activity.
3. In this application, the activity implements Constraint Layout, Linear Layout, and Relative Layout
4. There are various event listeners like onclick, ondoubleclick, onlongPress, onTouch events. In this activity onclick is implemented.

**Code:**

**Activity\_main.xml:**

<?xml version="1.0" encoding="utf-8"?>

<RelativeLayout xmlns:androi[d="http://schemas.android.com/apk/res/android"](http://schemas.android.com/apk/res/android) xmlns:tools="<http://schemas.android.com/tools>" android:layout\_width="match\_parent" android:layout\_height="match\_parent"

tools:context=".MainActivity">

<LinearLayout android:layout\_width="match\_parent" android:layout\_height="100dp">

<TextView android:id="@+id/textView"

android:layout\_width="match\_parent"

android:layout\_height="wrap\_content" android:layout\_margin="10dp" android:text="Details Form" android:textSize="15sp" android:gravity="center"/>

</LinearLayout>

<GridLayout android:id="@+id/gridLayout" android:layout\_width="match\_parent" android:layout\_height="match\_parent" android:layout\_marginTop="100dp" android:layout\_marginBottom="100dp" android:columnCount="2" android:rowCount="3">

<TextView android:id="@+id/textView1" android:layout\_width="wrap\_content" android:layout\_height="wrap\_content" android:layout\_margin="10dp" android:layout\_row="0" android:layout\_column="0" android:text="Name" android:textSize="10sp" android:gravity="center"/>

<EditText android:id="@+id/editText"

android:layout\_width="wrap\_content" android:layout\_height="wrap\_content" android:layout\_margin="10dp" android:layout\_row="0" android:layout\_column="1" android:ems="10"/>

<TextView android:id="@+id/textView2" android:layout\_width="wrap\_content" android:layout\_height="wrap\_content" android:layout\_margin="10dp" android:layout\_row="1" android:layout\_column="0" android:text="Reg.No"

android:textSize="10sp" android:gravity="center"/>

<EditText android:id="@+id/editText2"

android:layout\_width="wrap\_content" android:layout\_height="wrap\_content" android:layout\_margin="10dp" android:layout\_row="1" android:layout\_column="1" android:inputType="number" android:ems="10"/>

<TextView android:id="@+id/textView3" android:layout\_width="wrap\_content" android:layout\_height="wrap\_content" android:layout\_margin="10dp" android:layout\_row="2" android:layout\_column="0" android:text="Dept" android:textSize="20sp" android:gravity="center"/>

<Spinner android:id="@+id/spinner"

android:layout\_width="wrap\_content" android:layout\_height="wrap\_content" android:layout\_margin="10dp" android:layout\_row="2" android:layout\_column="1" android:spinnerMode="dropdown"/>

</GridLayout>

<Button

android:id="@+id/button" android:layout\_width="wrap\_content" android:layout\_height="wrap\_content" android:layout\_alignParentBottom="true" android:layout\_centerInParent="true" android:layout\_marginBottom="80dp" android:text="Submit"/>

</RelativeLayout>

## Activity\_second.xml:

<?xml version="1.0" encoding="utf-8"?>

<LinearLayout xmlns:androi[d="http://schemas.android.com/apk/res/android"](http://schemas.android.com/apk/res/android) xmlns:tools="<http://schemas.android.com/tools>" android:layout\_width="match\_parent" android:layout\_height="match\_parent" tools:context="com.example.devang.exno2.SecondActivity" android:orientation="vertical"

android:gravity="center">

<TextView android:id="@+id/textView1" android:layout\_width="wrap\_content" android:layout\_height="wrap\_content" android:layout\_margin="20dp" android:text="New Text" android:textSize="30sp"/>

<TextView android:id="@+id/textView2" android:layout\_width="wrap\_content" android:layout\_height="wrap\_content" android:layout\_margin="20dp" android:text="New Text" android:textSize="30sp"/>

<TextView android:id="@+id/textView3" android:layout\_width="wrap\_content" android:layout\_height="wrap\_content" android:layout\_margin="20dp" android:text="New Text" android:textSize="30sp"/>

</LinearLayout> **SecondActivity.java:** package com.example.exno2;

import android.content.Intent;

import androidx.appcompat.app.AppCompatActivity; import android.os.Bundle;

import android.widget.TextView;

public class SecondActivity extends AppCompatActivity { TextView t1,t2,t3;

String name,reg,dept;

@Override

protected void onCreate(Bundle savedInstanceState) { super.onCreate(savedInstanceState); setContentView(R.layout.activity\_second);

t1= (TextView) findViewById(R.id.textView1); t2= (TextView) findViewById(R.id.textView2); t3= (TextView) findViewById(R.id.textView3);

//Getting the Intent Intent i = getIntent();

//Getting the Values from First Activity using the Intent received name=i.getStringExtra("name\_key"); reg=i.getStringExtra("reg\_key"); dept=i.getStringExtra("dept\_key");

//Setting the Values to Intent t1.setText(name); t2.setText(reg); t3.setText(dept);

}

}

## MainActivity.java:

package com.example.exno2; import android.content.Intent;

import androidx.appcompat.app.AppCompatActivity;

import android.os.Bundle; import android.view.View;

import android.widget.ArrayAdapter; import android.widget.Button; import android.widget.EditText; import android.widget.Spinner;

public class MainActivity extends AppCompatActivity {

//Defining the Views EditText e1,e2; Button bt;

Spinner s;

//Data for populating in Spinner

String [] dept\_array={"CSE","ECE","IT","Mech","Civil"};

String name,reg,dept; @Override

protected void onCreate(Bundle savedInstanceState) { super.onCreate(savedInstanceState); setContentView(R.layout.activity\_main);

//Referring the Views

e1= (EditText) findViewById(R.id.editText); e2= (EditText) findViewById(R.id.editText2);

bt= (Button) findViewById(R.id.button); s= (Spinner) findViewById(R.id.spinner);

//Creating Adapter for Spinner for adapting the data from array to Spinner ArrayAdapter adapter= new

ArrayAdapter(MainActivity.this,android.R.layout.simple\_spinner\_item,dept\_array); s.setAdapter(adapter);

//Creating Listener for Button bt.setOnClickListener(new View.OnClickListener() {

@Override

public void onClick(View v) {

//Getting the Values from Views(Edittext & Spinner) name=e1.getText().toString(); reg=e2.getText().toString();

dept=s.getSelectedItem().toString();

//Intent For Navigating to Second Activity

Intent i = new Intent(MainActivity.this,SecondActivity.class);

//For Passing the Values to Second Activity i.putExtra("name\_key", name); i.putExtra("reg\_key",reg); i.putExtra("dept\_key", dept);

startActivity(i);

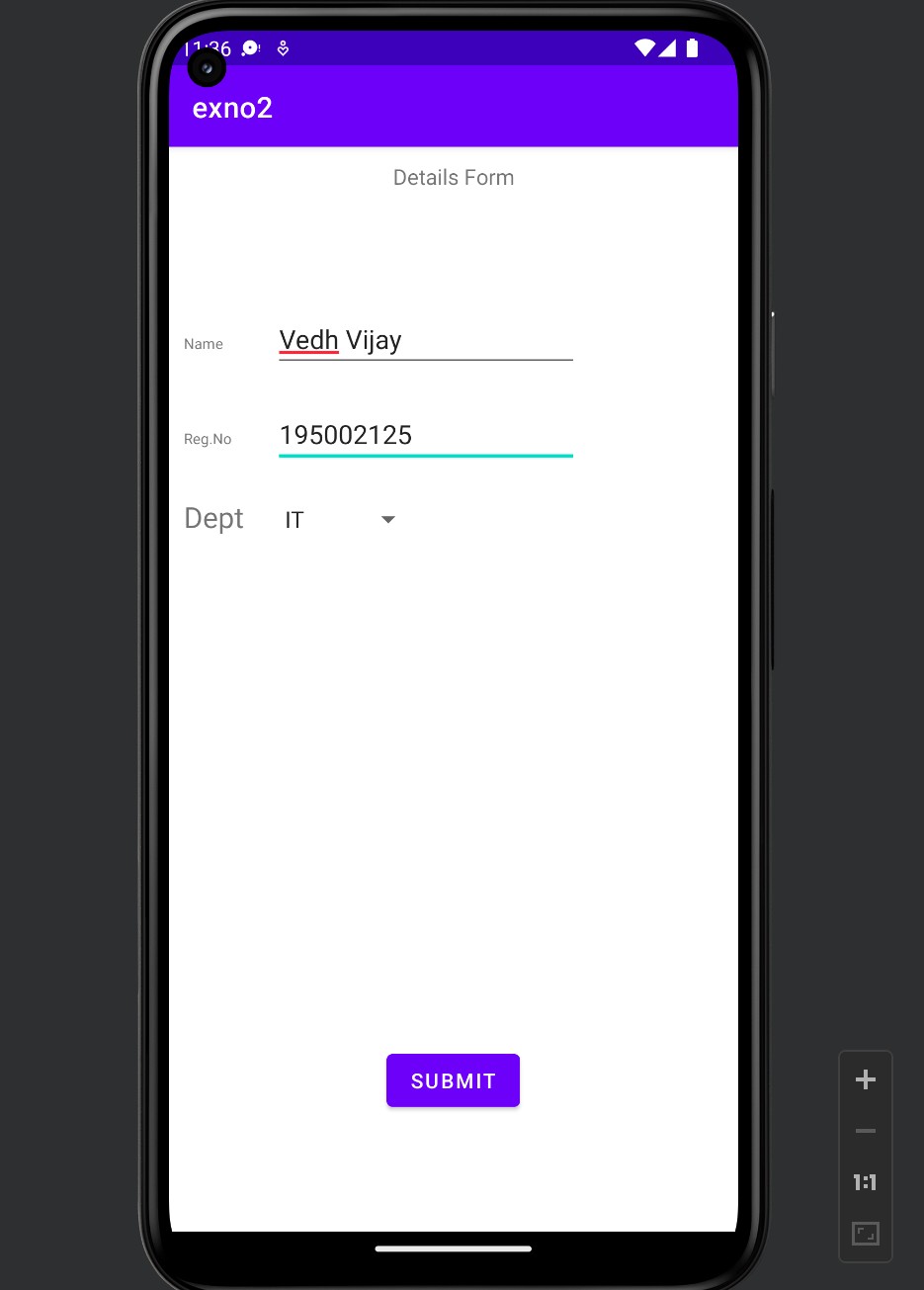
}

});

}

}

## Sample I/O:



195003305

Shrishti Sinha

**Result:**

Thus a Simple Android Application that uses Layout Managers and Event Listeners is developed and executed successfully.

**Ex.No: 3**

**Simple Calculator Application**

Date: 13/09/2022

**Aim:**

To implement a simple calculator in android studio.

**Software and Hardware Requirements:**

Software:

1. Android Studio
2. Java Hardware:
3. Preferably 8GB+ RAM Laptop/Desktop

**Algorithm:**

* 1. Open Android Studio and then click on File -> New -> New project.
  2. Then type the Application name as “calculator″ and click Next.
  3. Then select the Empty Activity and click Next.
  4. Finally click Finish.
  5. The buttons required to implement the calculator are provided in the xml file.
  6. The implementation of functions for the respective button clicks are specified in the java file.
  7. When equal button is pressed the function implementation is done

**Code:**

**Activity\_main.xml:**

<?xml version="1.0" encoding="utf-8"?>

<LinearLayout xmlns:androi[d="http://schemas.android.com/apk/res/android"](http://schemas.android.com/apk/res/android) android:orientation="vertical" android:layout\_width="match\_parent" android:layout\_height="match\_parent" android:layout\_margin="20dp">

<LinearLayout

android:id="@+id/linearLayout1" android:layout\_width="match\_parent" android:layout\_height="wrap\_content" android:layout\_margin="20dp">

<EditText android:id="@+id/editText1"

android:layout\_width="match\_parent" android:layout\_height="wrap\_content" android:layout\_weight="1" android:inputType="numberDecimal" android:textSize="20sp" />

<EditText android:id="@+id/editText2"

android:layout\_width="match\_parent" android:layout\_height="wrap\_content" android:layout\_weight="1" android:inputType="numberDecimal" android:textSize="20sp" />

</LinearLayout>

<LinearLayout android:id="@+id/linearLayout2" android:layout\_width="match\_parent" android:layout\_height="wrap\_content" android:layout\_margin="20dp">

<Button

android:id="@+id/Add" android:layout\_width="match\_parent" android:layout\_height="wrap\_content" android:layout\_weight="1" android:text="+" android:textSize="30sp"/>

<Button

android:id="@+id/Sub" android:layout\_width="match\_parent" android:layout\_height="wrap\_content" android:layout\_weight="1" android:text="-"

android:textSize="30sp"/>

<Button

android:id="@+id/Mul" android:layout\_width="match\_parent" android:layout\_height="wrap\_content" android:layout\_weight="1" android:text="\*" android:textSize="30sp"/>

<Button

android:id="@+id/Div" android:layout\_width="match\_parent" android:layout\_height="wrap\_content" android:layout\_weight="1" android:text="/" android:textSize="30sp"/>

</LinearLayout>

<TextView android:id="@+id/textView"

android:layout\_width="match\_parent" android:layout\_height="wrap\_content" android:layout\_marginTop="50dp" android:text="Answer is" android:textSize="30sp" android:gravity="center"/>

</LinearLayout>

## MainActivity\_java:

package com.example.calculator;

import android.os.Bundle; import android.text.TextUtils; import android.view.View;

import android.view.View.OnClickListener; import android.widget.Button;

import android.widget.EditText; import android.widget.TextView;

import androidx.appcompat.app.AppCompatActivity;

public class MainActivity extends AppCompatActivity implements OnClickListener

{

//Defining the Views EditText Num1; EditText Num2; Button Add;

Button Sub;

Button Mul;

Button Div;

TextView Result;

@Override

public void onCreate(Bundle savedInstanceState)

{

super.onCreate(savedInstanceState); setContentView(R.layout.activity\_main);

//Referring the Views

Num1 = (EditText) findViewById(R.id.editText1); Num2 = (EditText) findViewById(R.id.editText2); Add = (Button) findViewById(R.id.Add);

Sub = (Button) findViewById(R.id.Sub); Mul = (Button) findViewById(R.id.Mul); Div = (Button) findViewById(R.id.Div);

Result = (TextView) findViewById(R.id.textView);

// set a listener Add.setOnClickListener(this); Sub.setOnClickListener(this); Mul.setOnClickListener(this); Div.setOnClickListener(this);

}

@Override

public void onClick (View v)

{

float num1 = 0; float num2 = 0; float result = 0; String oper = "";

// check if the fields are empty

if (TextUtils.isEmpty(Num1.getText().toString()) || TextUtils.isEmpty(Num2.getText().toString()))

return;

// read EditText and fill variables with numbers num1 = Float.parseFloat(Num1.getText().toString()); num2 = Float.parseFloat(Num2.getText().toString());

// defines the button that has been clicked and performs the corresponding operation

// write operation into oper, we will use it later for output switch (v.getId())

{

case R.id.Add: oper = "+";

result = num1 + num2; break;

case R.id.Sub: oper = "-";

result = num1 - num2; break;

case R.id.Mul: oper = "\*";

result = num1 \* num2; break;

case R.id.Div: oper = "/";

result = num1 / num2; break;

default:

break;

}

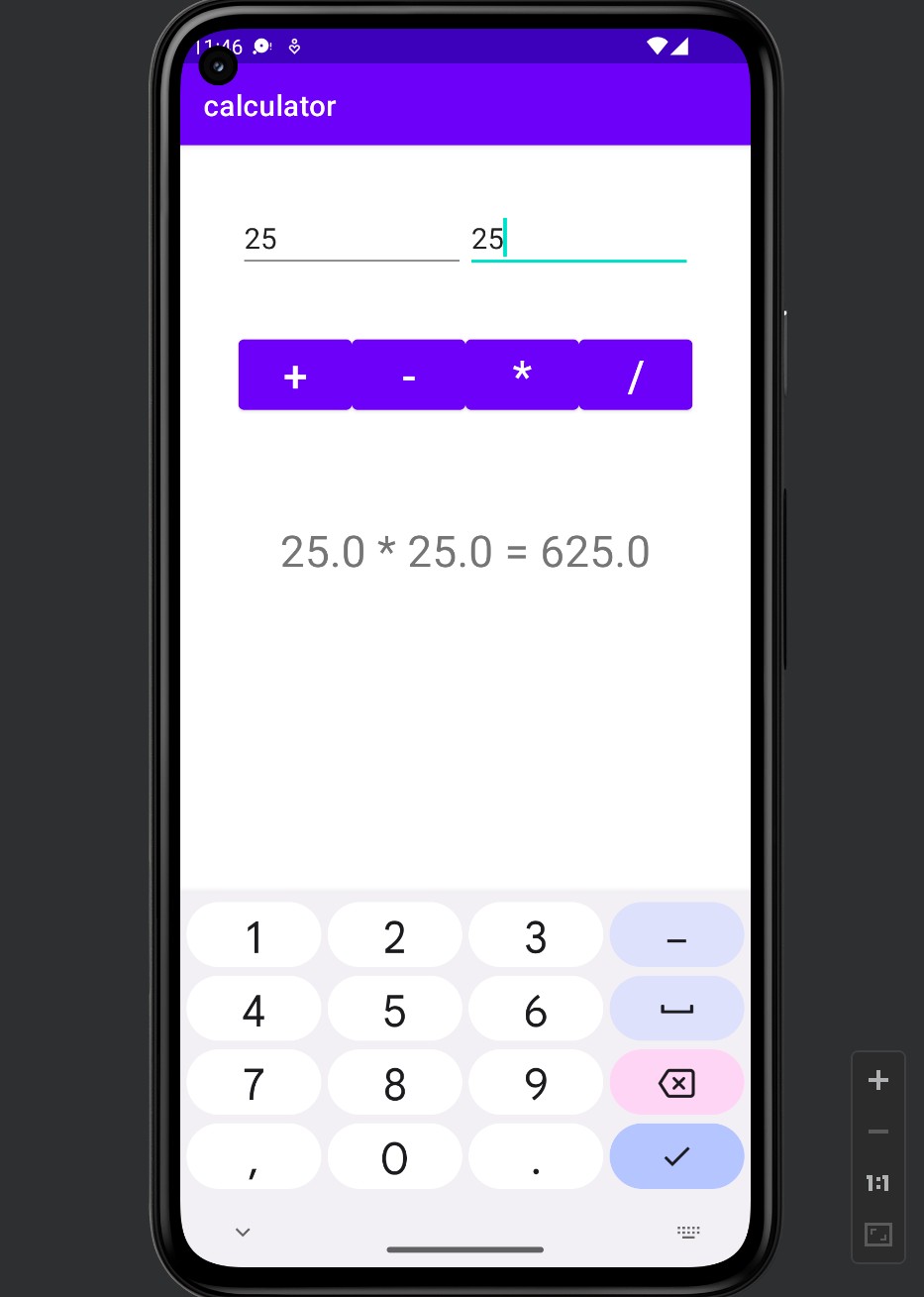
// form the output line

Result.setText(num1 + " " + oper + " " + num2 + " = " + result);

}

}

## Sample I/O:



**Result:**

Thus a Simple Android Application for Native Calculator is developed and executed successfully.

**Basic Graphical Primitives**

**Ex.No: 4**

Date: 20/09/2022

**Aim:**

To implement basic graphical primitives in android studio.

**Software and Hardware Requirements:**

Software:

1. Android Studio
2. Java Hardware:

1. Preferably 8GB+ RAM Laptop/Desktop

**Algorithm:**

1. Open Android Studio and create a new project.
2. Select Empty Activity.
3. Android provides a huge set of 2D-drawing APIs that allow you to create graphics.
4. The Bitmap (android.graphics.Bitmap) class represents a bitmap image.
5. A Canvas is an object that you can draw on by calling drawing commands. In this program, we have drawn a rectangle, circle, square and line using Canvas class.

**Code:**

**Activity\_main.xml:**

<?xml version="1.0" encoding="utf-8"?>

<RelativeLayout xmlns:androi[d="http://schemas.android.com/apk/res/android"](http://schemas.android.com/apk/res/android) android:layout\_width="match\_parent" android:layout\_height="match\_parent">

<ImageView android:layout\_width="match\_parent" android:layout\_height="match\_parent" android:id="@+id/imageView" />

</RelativeLayout> **MainActivity.java:** package com.example.graphics;

import android.app.Activity; import android.graphics.Bitmap; import android.graphics.Canvas; import android.graphics.Color; import android.graphics.Paint;

import android.graphics.drawable.BitmapDrawable; import android.os.Bundle;

import android.widget.ImageView;

public class MainActivity extends Activity

{

@Override

public void onCreate(Bundle savedInstanceState)

{

super.onCreate(savedInstanceState); setContentView(R.layout.activity\_main);

//Creating a Bitmap

Bitmap bg = Bitmap.createBitmap(720, 1280, Bitmap.Config.ARGB\_8888);

//Setting the Bitmap as background for the ImageView ImageView i = (ImageView) findViewById(R.id.imageView); i.setBackgroundDrawable(new BitmapDrawable(bg));

//Creating the Canvas Object Canvas canvas = new Canvas(bg);

//Creating the Paint Object and set its color & TextSize Paint paint = new Paint(); paint.setColor(Color.GREEN);

paint.setTextSize(50);

//To draw a Rectangle canvas.drawText("Rectangle", 420, 150, paint);

canvas.drawRect(400, 200, 650, 700, paint);

//To draw a Circle canvas.drawText("Circle", 120, 150, paint);

canvas.drawCircle(200, 350, 150, paint);

//To draw a Square canvas.drawText("Square", 120, 800, paint);

canvas.drawRect(50, 850, 350, 1150, paint);

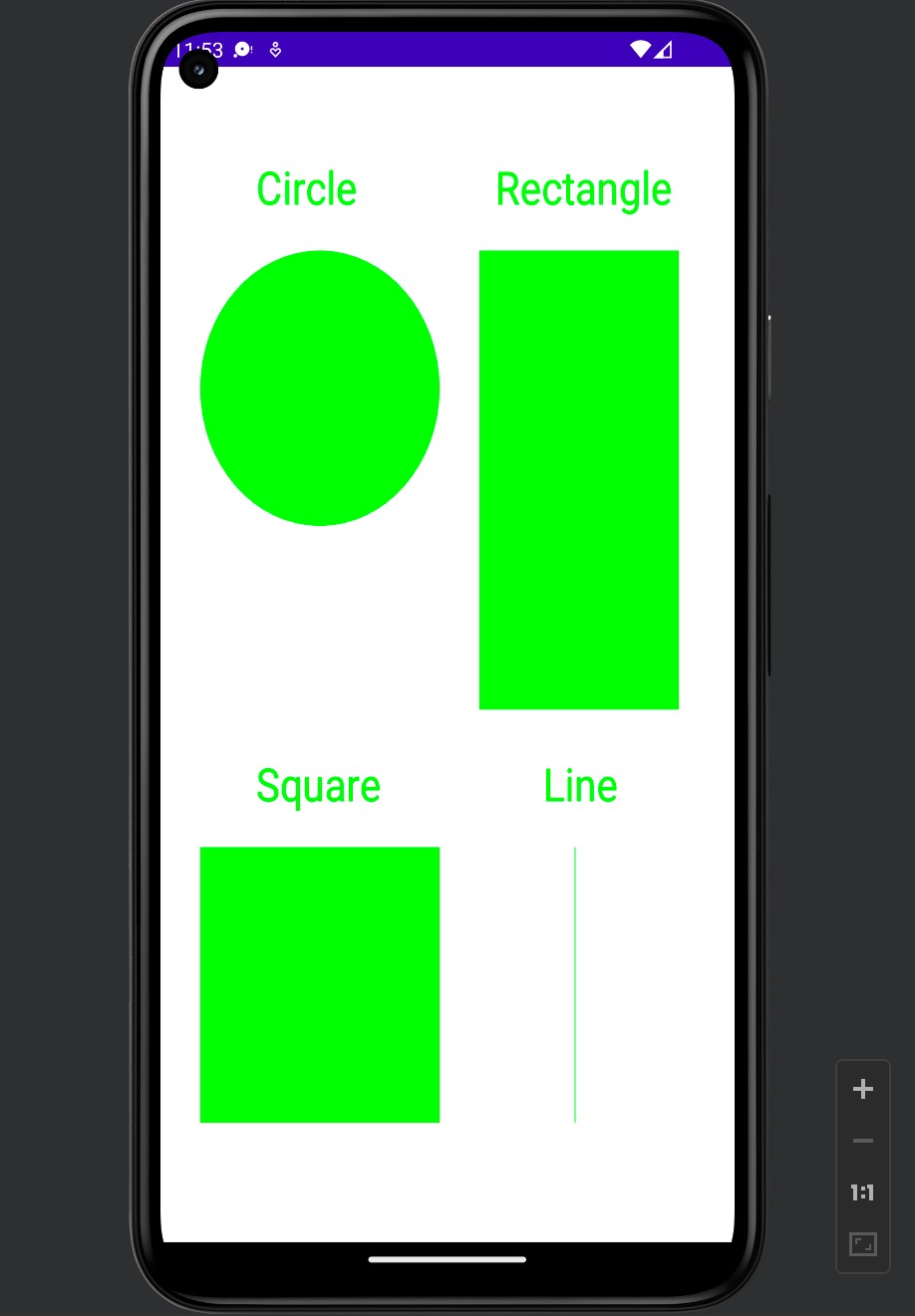
//To draw a Line canvas.drawText("Line", 480, 800, paint);

canvas.drawLine(520, 850, 520, 1150, paint);

}

}

## Sample I/O:



**Result:**

Thus a Simple Android Application that draws basic Graphical Primitives on the screen is developed and executed successfully.

**Database**

Date: 27/09/2022

**Ex.No: 5**

**Aim:**

To develop a simple Android Application using a database.

**Software and Hardware Requirements:**

Software:

* 1. Android Studio
  2. Java Hardware:

1. Preferably 8GB+ RAM Laptop/Desktop

**Algorithm:**

* 1. Open Android Studio and create a new project.
  2. Select Empty Activity.
  3. SQLiteDatabase has methods to create, delete, execute SQL commands, and perform other common database management tasks.
  4. The Builder class has several methods. This program makes use of the following:
     + setCancelable(boolean flag): Sets whether this dialog is cancelable with the BACK key.
     + setTitle(char Sequence): Makes the string passed as the title.
  5. Event Listeners are used to track mouse clicks

**Code:**

**MainActivity.java**

package com.example.database; import android.os.Bundle;

import android.app.Activity;

import android.app.AlertDialog.Builder; import android.content.Context;

import android.database.Cursor;

import android.database.sqlite.SQLiteDatabase;

import android.os.Bundle; import android.view.View;

import android.view.View.OnClickListener; import android.widget.Button;

import android.widget.EditText;

public class MainActivity extends Activity implements OnClickListener

{

EditText Rollno,Name,Marks;

Button Insert,Delete,Update,View,ViewAll; SQLiteDatabase db;

/\*\* Called when the activity is first created. \*/ @Override

public void onCreate(Bundle savedInstanceState)

{

super.onCreate(savedInstanceState); setContentView(R.layout.activity\_main);

Rollno=(EditText)findViewById(R.id.Rollno); Name=(EditText)findViewById(R.id.Name); Marks=(EditText)findViewById(R.id.Marks); Insert=(Button)findViewById(R.id.Insert); Delete=(Button)findViewById(R.id.Delete); Update=(Button)findViewById(R.id.Update); View=(Button)findViewById(R.id.View); ViewAll=(Button)findViewById(R.id.ViewAll);

Insert.setOnClickListener(this); Delete.setOnClickListener(this); Update.setOnClickListener(this); View.setOnClickListener(this); ViewAll.setOnClickListener(this);

// Creating database and table

db=openOrCreateDatabase("StudentDB", Context.MODE\_PRIVATE, null); db.execSQL("CREATE TABLE IF NOT EXISTS student(rollno VARCHAR,name

VARCHAR,marks VARCHAR);");

}

public void onClick(View view)

{

// Inserting a record to the Student table if(view==Insert)

{

// Checking for empty fields if(Rollno.getText().toString().trim().length()==0||

Name.getText().toString().trim().length()==0|| Marks.getText().toString().trim().length()==0)

{

showMessage("Error", "Please enter all values"); return;

}

db.execSQL("INSERT INTO student VALUES('"+Rollno.getText()

+"','"+Name.getText()+

"','"+Marks.getText()+"');"); showMessage("Success", "Record added"); clearText();

}

// Deleting a record from the Student table if(view==Delete)

{

// Checking for empty roll number if(Rollno.getText().toString().trim().length()==0)

{

showMessage("Error", "Please enter Rollno"); return;

}

Cursor c=db.rawQuery("SELECT \* FROM student WHERE rollno='"+Rollno.getText()+"'", null);

if(c.moveToFirst())

{

db.execSQL("DELETE FROM student WHERE rollno='"+Rollno.getText()+"'"); showMessage("Success", "Record Deleted");

}

else

{

showMessage("Error", "Invalid Rollno");

}

clearText();

}

// Updating a record in the Student table if(view==Update)

{

// Checking for empty roll number if(Rollno.getText().toString().trim().length()==0)

{

showMessage("Error", "Please enter Rollno"); return;

}

Cursor c=db.rawQuery("SELECT \* FROM student WHERE rollno='"+Rollno.getText()+"'", null);

if(c.moveToFirst()) {

db.execSQL("UPDATE student SET name='" + Name.getText() + "',marks='" + Marks.getText() +

"' WHERE rollno='"+Rollno.getText()+"'"); showMessage("Success", "Record Modified");

}

else {

showMessage("Error", "Invalid Rollno");

}

clearText();

}

// Display a record from the Student table if(view==View)

{

// Checking for empty roll number if(Rollno.getText().toString().trim().length()==0)

{

showMessage("Error", "Please enter Rollno"); return;

}

Cursor c=db.rawQuery("SELECT \* FROM student WHERE rollno='"+Rollno.getText()+"'", null);

if(c.moveToFirst())

{

Name.setText(c.getString(1)); Marks.setText(c.getString(2));

}

else

{

showMessage("Error", "Invalid Rollno"); clearText();

}

}

// Displaying all the records if(view==ViewAll)

{

Cursor c=db.rawQuery("SELECT \* FROM student", null); if(c.getCount()==0)

{

showMessage("Error", "No records found"); return;

}

StringBuffer buffer=new StringBuffer(); while(c.moveToNext())

{

buffer.append("Rollno: "+c.getString(0)+"\n"); buffer.append("Name: "+c.getString(1)+"\n"); buffer.append("Marks: "+c.getString(2)+"\n\n");

}

showMessage("Student Details", buffer.toString());

}

}

public void showMessage(String title,String message)

{

Builder builder=new Builder(this); builder.setCancelable(true); builder.setTitle(title); builder.setMessage(message); builder.show();

}

public void clearText()

{

Rollno.setText("");

Name.setText("");

Marks.setText(""); Rollno.requestFocus();

}

}

## activity\_main.xml:

<?xml version="1.0" encoding="utf-8"?>

<AbsoluteLayout xmlns:androi[d="http://schemas.android.com/apk/res/android"](http://schemas.android.com/apk/res/android) android:layout\_width="match\_parent" android:layout\_height="match\_parent">

<TextView android:layout\_width="wrap\_content" android:layout\_height="wrap\_content" android:layout\_x="50dp" android:layout\_y="20dp" android:text="Student Details" android:textSize="30sp" />

<TextView android:layout\_width="wrap\_content" android:layout\_height="wrap\_content" android:layout\_x="20dp" android:layout\_y="110dp" android:text="Enter Rollno:" android:textSize="20sp" />

<EditText android:id="@+id/Rollno" android:layout\_width="150dp"

android:layout\_height="wrap\_content" android:layout\_x="175dp" android:layout\_y="100dp" android:inputType="number" android:textSize="20sp" />

<TextView android:layout\_width="wrap\_content" android:layout\_height="wrap\_content" android:layout\_x="20dp" android:layout\_y="160dp" android:text="Enter Name:" android:textSize="20sp" />

<EditText android:id="@+id/Name" android:layout\_width="150dp"

android:layout\_height="wrap\_content" android:layout\_x="175dp" android:layout\_y="150dp" android:inputType="text" android:textSize="20sp" />

<TextView android:layout\_width="wrap\_content" android:layout\_height="wrap\_content" android:layout\_x="20dp" android:layout\_y="210dp" android:text="Enter Marks:" android:textSize="20sp" />

<EditText android:id="@+id/Marks" android:layout\_width="150dp"

android:layout\_height="wrap\_content" android:layout\_x="175dp" android:layout\_y="200dp" android:inputType="number" android:textSize="20sp" />

<Button

android:id="@+id/Insert" android:layout\_width="150dp"

android:layout\_height="wrap\_content" android:layout\_x="25dp" android:layout\_y="300dp" android:text="Insert" android:textSize="30dp" />

<Button

android:id="@+id/Delete" android:layout\_width="150dp" android:layout\_height="wrap\_content" android:layout\_x="200dp" android:layout\_y="300dp" android:text="Delete" android:textSize="30dp" />

<Button

android:id="@+id/Update" android:layout\_width="150dp" android:layout\_height="wrap\_content" android:layout\_x="25dp" android:layout\_y="400dp" android:text="Update" android:textSize="30dp" />

<Button

android:id="@+id/View" android:layout\_width="150dp" android:layout\_height="wrap\_content" android:layout\_x="200dp" android:layout\_y="400dp" android:text="View" android:textSize="30dp" />

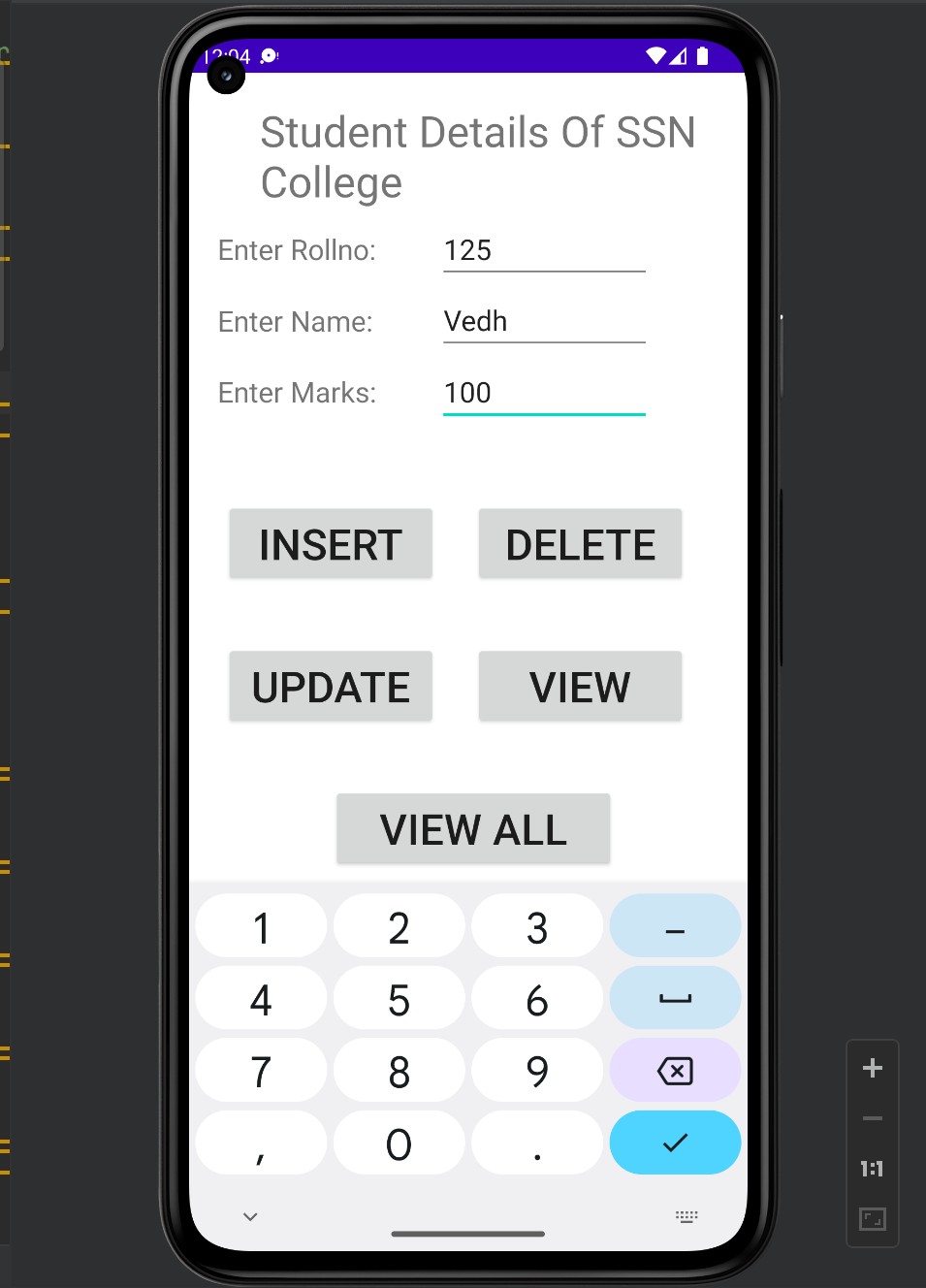
<Button

android:id="@+id/ViewAll" android:layout\_width="200dp" android:layout\_height="wrap\_content" android:layout\_x="100dp"

android:layout\_y="500dp" android:text="View All" android:textSize="30dp" />

</AbsoluteLayout>

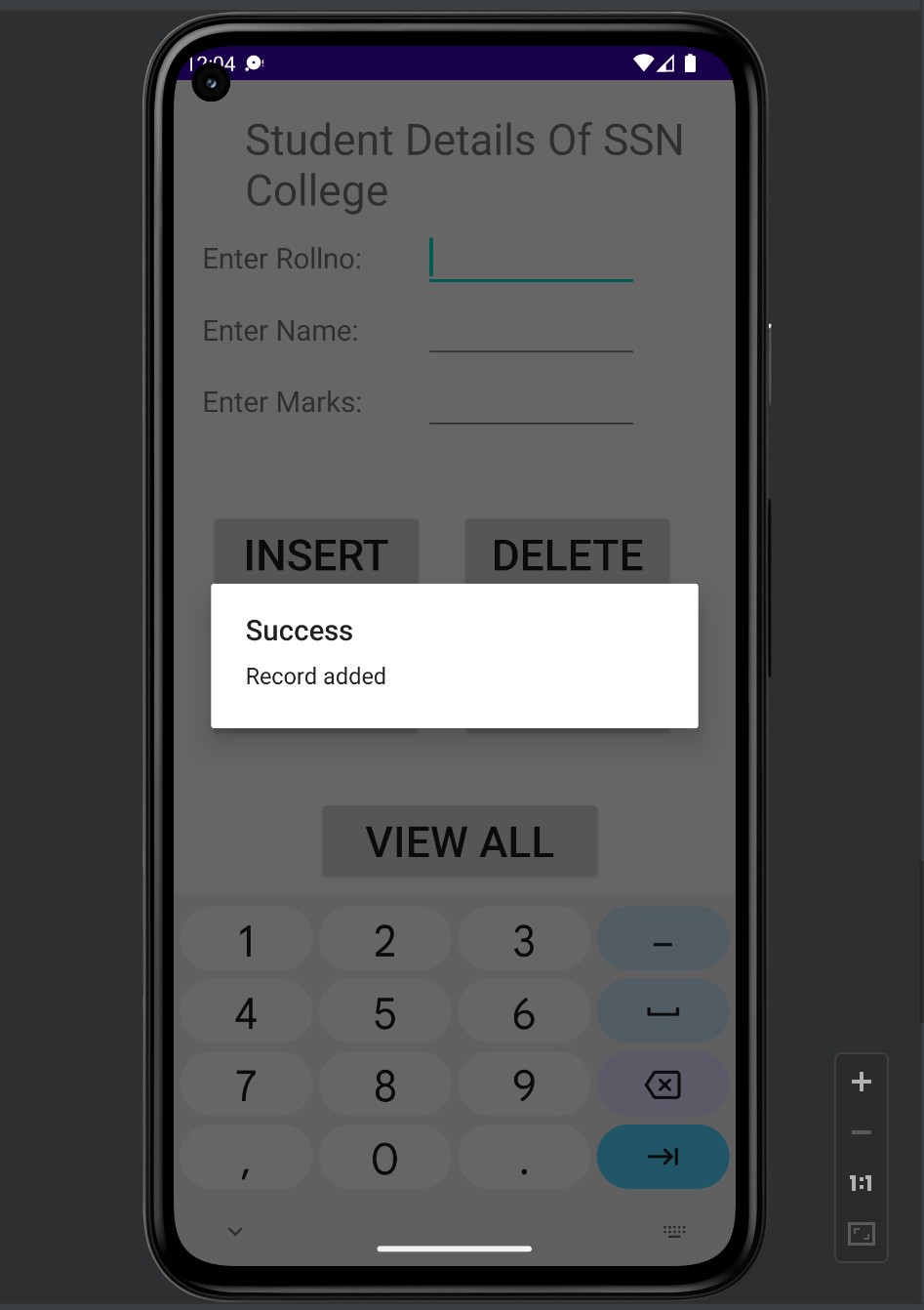
## Sample I/O:



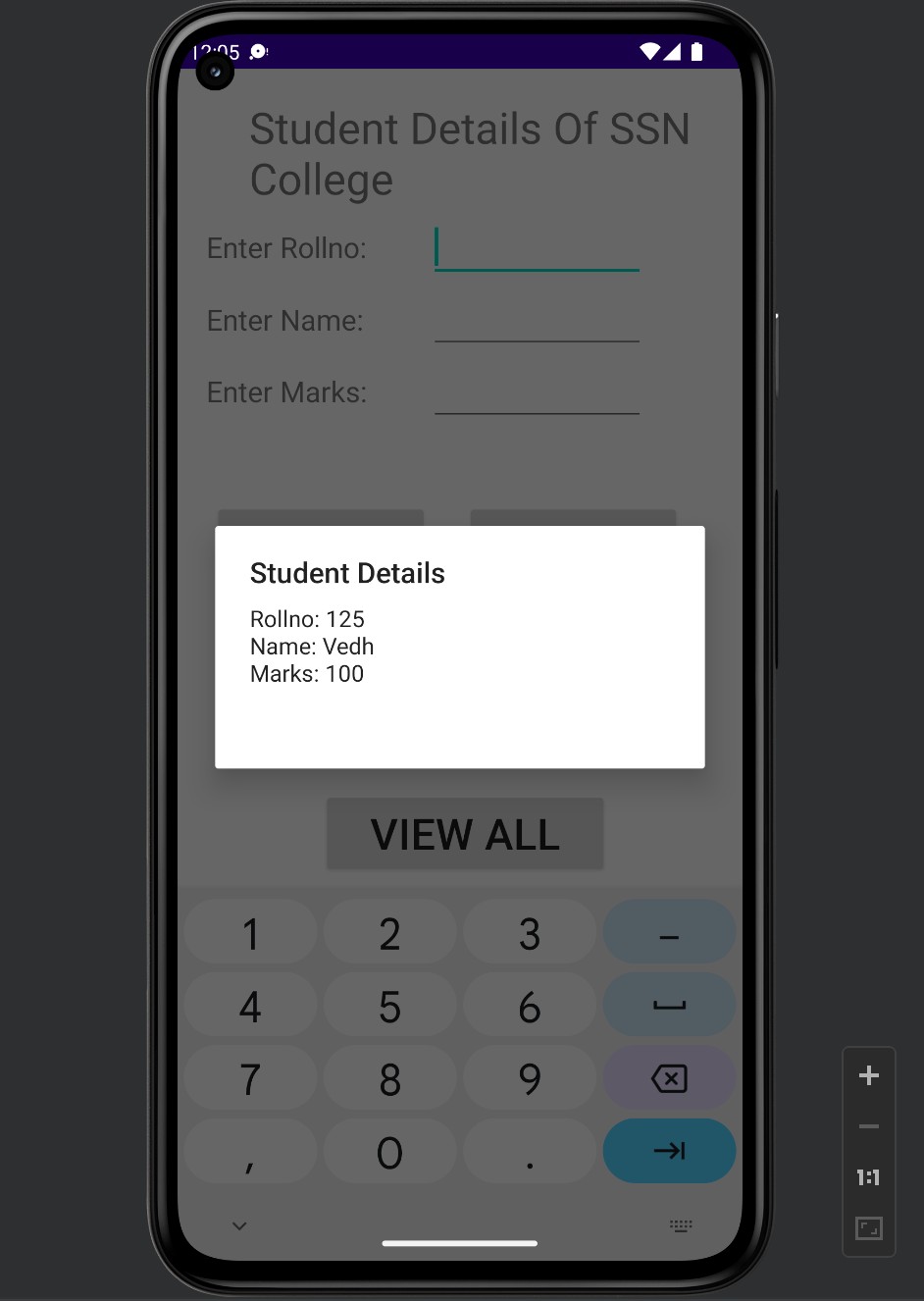
Shrishti

1195

**Insert:**



**View:**



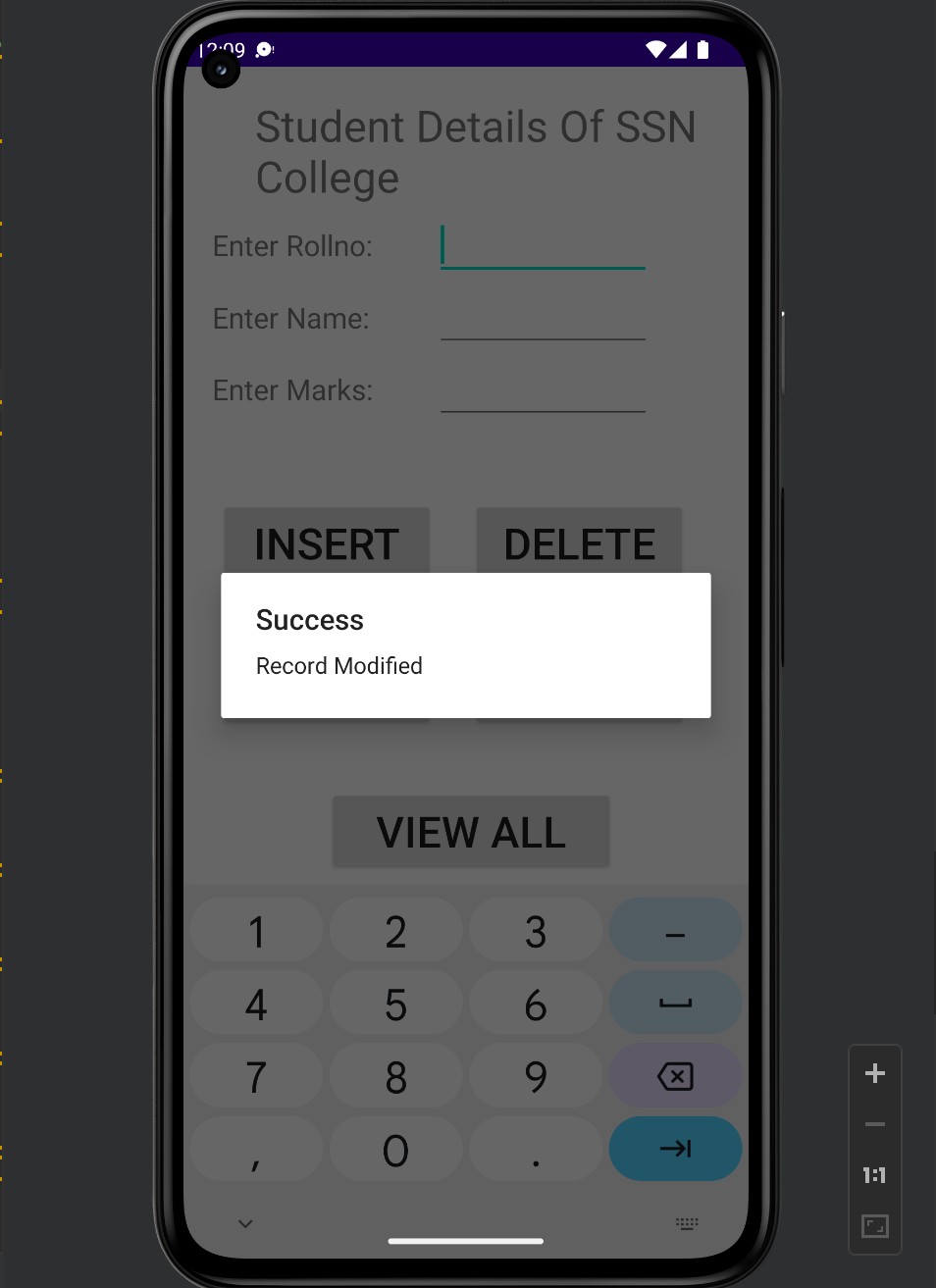
1195

Shrishti

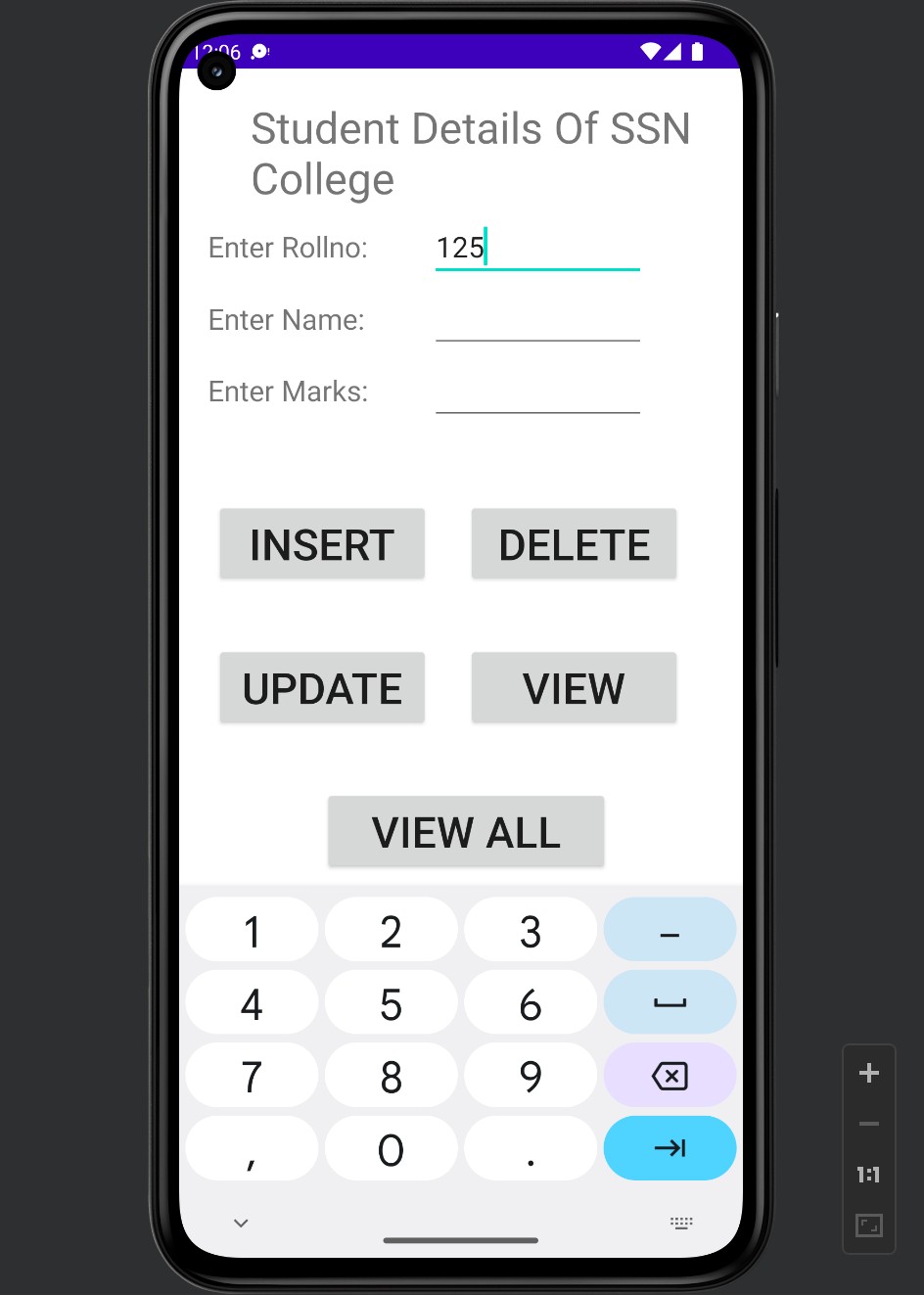
100

­

**Update**:

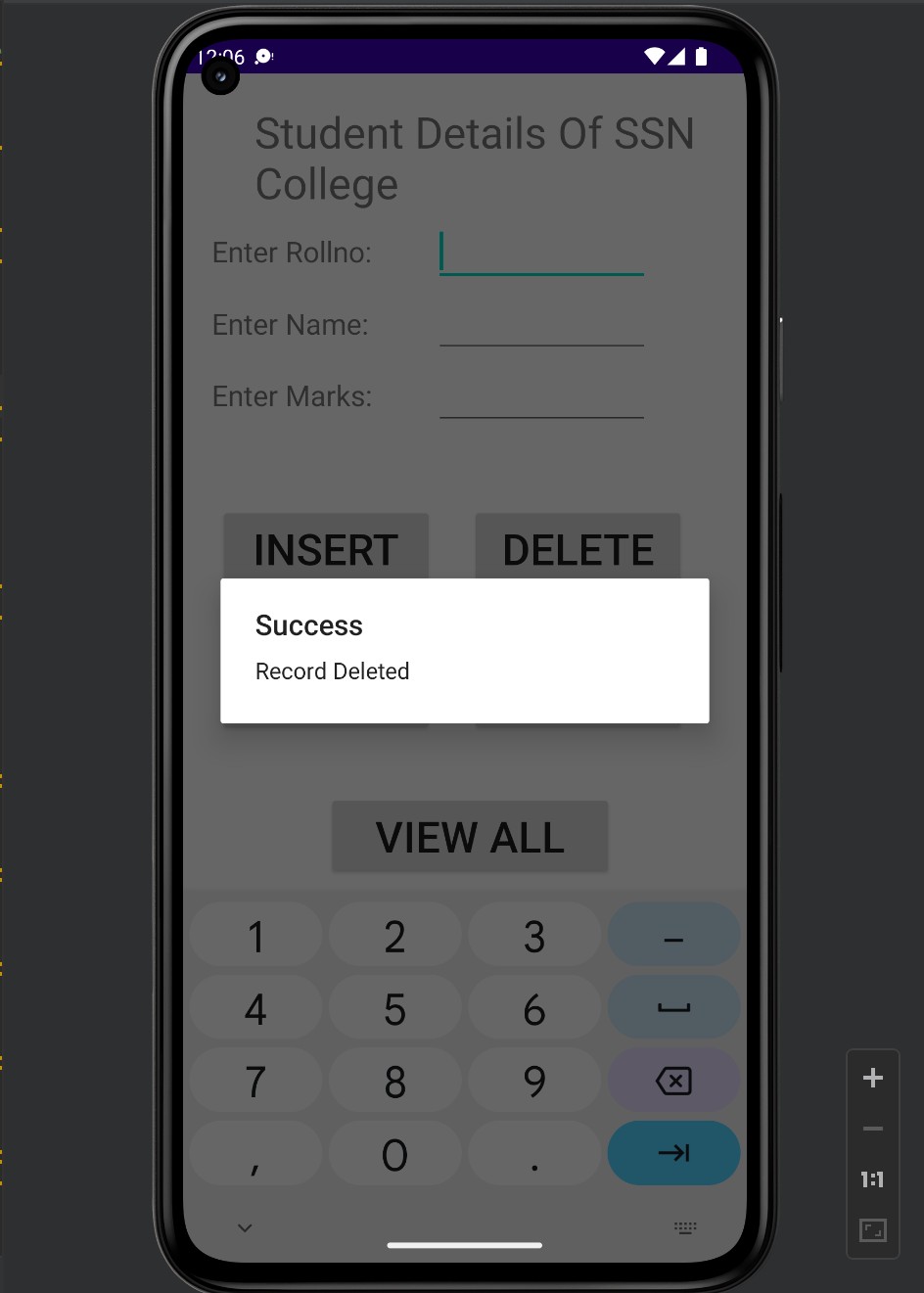


**Deletion**:



1195

­



**Result:**

Thus, a simple application using a database has been successfully implemented using android studio.

**RSS**

**Ex.No: 6**

Date: 11/10/2022

**Aim:**

To develop an Android application for RSS (Really Simple Syndication) Feed using Android Studio.

**Software and Hardware Requirements:**

Software:

1. Android Studio
2. Java Hardware:
   1. Preferably 8GB+ RAM Laptop/Desktop

**Algorithm:**

1. Open Android Studio and create a new project.
2. Select Empty Activity.
3. The Main.java file makes use of XMLPullParser that parses through the RSS XML file.
4. Add the following in android manifest xml file

<**uses-permission android:name="android.permission.INTERNET"**/>

## Code: MainActivity.java

package com.example.rss;

import android.app.ListActivity; import android.content.Intent; import android.net.Uri;

import android.os.AsyncTask; import android.os.Bundle; import android.view.View;

import android.widget.ArrayAdapter; import android.widget.ListView; import org.xmlpull.v1.XmlPullParser;

import org.xmlpull.v1.XmlPullParserException; import org.xmlpull.v1.XmlPullParserFactory; import java.io.IOException;

import java.io.InputStream;

import java.net.MalformedURLException; import java.net.URL;

import java.util.ArrayList; import java.util.List;

public class MainActivity extends ListActivity

{

List headlines; List links;

@Override

protected void onCreate(Bundle savedInstanceState)

{

super.onCreate(savedInstanceState); new MyAsyncTask().execute();

}

class MyAsyncTask extends AsyncTask<Object,Void,ArrayAdapter>

{

@Override

protected ArrayAdapter doInBackground(Object[] params)

{

headlines = new ArrayList(); links = new ArrayList();

try

{

URL url = new URL("https://codingconnect.net/feed"); XmlPullParserFactory factory = XmlPullParserFactory.newInstance(); factory.setNamespaceAware(false);

XmlPullParser xpp = factory.newPullParser();

// We will get the XML from an input stream xpp.setInput(getInputStream(url), "UTF\_8"); boolean insideItem = false;

// Returns the type of current event: START\_TAG, END\_TAG, etc.. int eventType = xpp.getEventType();

while (eventType != XmlPullParser.END\_DOCUMENT)

{

if (eventType == XmlPullParser.START\_TAG)

{

if (xpp.getName().equalsIgnoreCase("item"))

{

insideItem = true;

}

else if (xpp.getName().equalsIgnoreCase("title"))

{

if (insideItem)

headlines.add(xpp.nextText()); //extract the headline

}

else if (xpp.getName().equalsIgnoreCase("link"))

{

if (insideItem)

links.add(xpp.nextText()); //extract the link of article

}

}

else if(eventType==XmlPullParser.END\_TAG && xpp.getName().equalsIgnoreCase("item"))

{

insideItem=false;

}

eventType = xpp.next(); //move to next element

}

}

catch (MalformedURLException e)

{

e.printStackTrace();

}

catch (XmlPullParserException e)

{

e.printStackTrace();

}

catch (IOException e)

{

e.printStackTrace();

}

return null;

}

protected void onPostExecute(ArrayAdapter adapter)

{

adapter = new ArrayAdapter(MainActivity.this, android.R.layout.simple\_list\_item\_1, headlines);

setListAdapter(adapter);

}

}

@Override

protected void onListItemClick(ListView l, View v, int position, long id)

{

Uri uri = Uri.parse((links.get(position)).toString()); Intent intent = new Intent(Intent.ACTION\_VIEW, uri); startActivity(intent);

}

public InputStream getInputStream(URL url)

{

try

{

return url.openConnection().getInputStream();

}

catch (IOException e)

{

return null;

}

}

}

## activity\_main.xml:

<?xml version="1.0" encoding="utf-8"?>

<LinearLayout xmlns:androi[d="http://schemas.android.com/apk/res/android"](http://schemas.android.com/apk/res/android) android:layout\_width="fill\_parent"

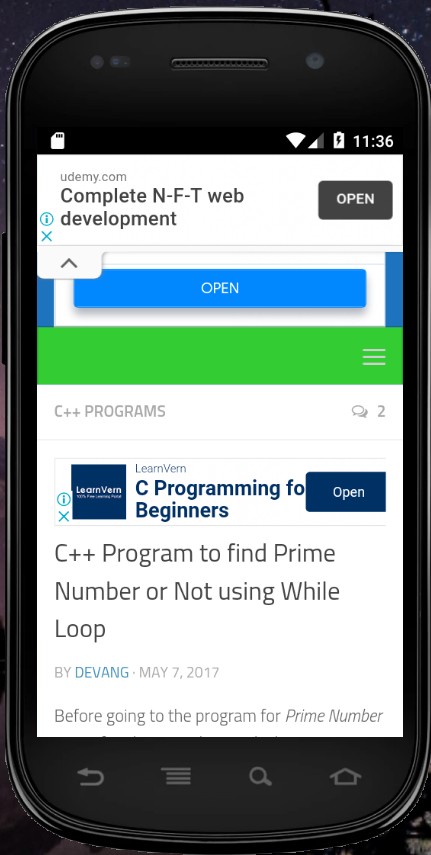
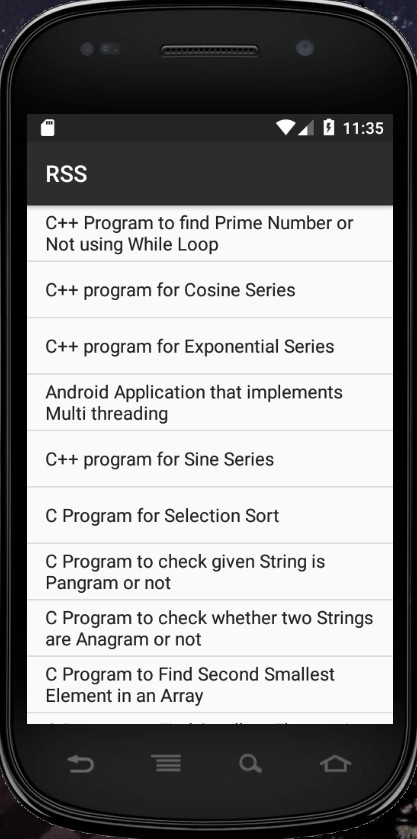
android:layout\_height="fill\_parent" android:orientation="vertical" >

<ListView android:id="@+id/listView"

android:layout\_width="match\_parent" android:layout\_height="wrap\_content" />

</LinearLayout>

## SAMPLE I/O:



**Result:**

Thus, an application for RSS (Really Simple Syndication) Feed using Android Studio is implemented successfully.

**Multithreading**

**Ex.No: 7**

Date: 18/10/2022

**Aim:**

To develop an Android Application that implements Multithreading.

**Software and Hardware Requirements:**

Software:

1. Android Studio
2. Java Hardware:
   1. Preferably 8GB+ RAM Laptop/Desktop

**Algorithm:**

1. A thread is a thread of execution in a program. The Java Virtual Machine allows an application to have multiple threads of execution running concurrently.
2. When a Java Virtual Machine starts up, there is usually a single non- daemon thread (which typically calls the method named main of some designated class). The Java Virtual Machine continues to execute threads until either of the following occurs:
   1. · The exit method of class Runtime has been called and the security manager has permitted the exit operation to take place.
   2. · All threads that are not daemon threads have died, either by returning from the call to the run method or by throwing an exception that propagates beyond the run method.
3. Using the runnable instance of the thread module, we multi-thread the process of loading two images of the India map.
4. On selecting the first button, one type of image will be displayed and on selecting the other button, the second image will be displayed immediately

**Code: MainActivity.java:**

package com.example.multithreading; import android.os.Bundle;

import androidx.appcompat.app.AppCompatActivity;

import android.view.View; import android.widget.Button;

import android.widget.ImageView;

public class MainActivity extends AppCompatActivity

{

ImageView img; Button bt1,bt2; @Override

protected void onCreate(Bundle savedInstanceState)

{

super.onCreate(savedInstanceState); setContentView(R.layout.activity\_main);

bt1 = (Button)findViewById(R.id.button); bt2= (Button) findViewById(R.id.button2);

img = (ImageView)findViewById(R.id.imageView);

bt1.setOnClickListener(new View.OnClickListener()

{

@Override

public void onClick(View v)

{

new Thread(new Runnable()

{

@Override public void run()

{

img.post(new Runnable()

{

});

}

@Override public void run()

{

img.setImageResource(R.drawable.pic1);

}

}

});

}).start();

bt2.setOnClickListener(new View.OnClickListener()

{

@Override

public void onClick(View v)

{

new Thread(new Runnable()

{

@Override public void run()

{

img.post(new Runnable()

{

@Override public void run()

{

img.setImageResource(R.drawable.pic2);

}

}

});

}

}

});

}

}).start();

## activitymain.xml:

<?xml version="1.0" encoding="utf-8"?>

<LinearLayout xmlns:androi[d="http://schemas.android.com/apk/res/android"](http://schemas.android.com/apk/res/android) android:layout\_width="match\_parent" android:layout\_height="match\_parent"

android:orientation="vertical" >

<ImageView android:id="@+id/imageView" android:layout\_width="250dp" android:layout\_height="250dp" android:layout\_margin="50dp" android:layout\_gravity="center" />

<Button android:id="@+id/button"

android:layout\_width="wrap\_content" android:layout\_height="wrap\_content" android:layout\_margin="5dp" android:layout\_gravity="center" android:text="Load Image 1" />

<Button android:id="@+id/button2"

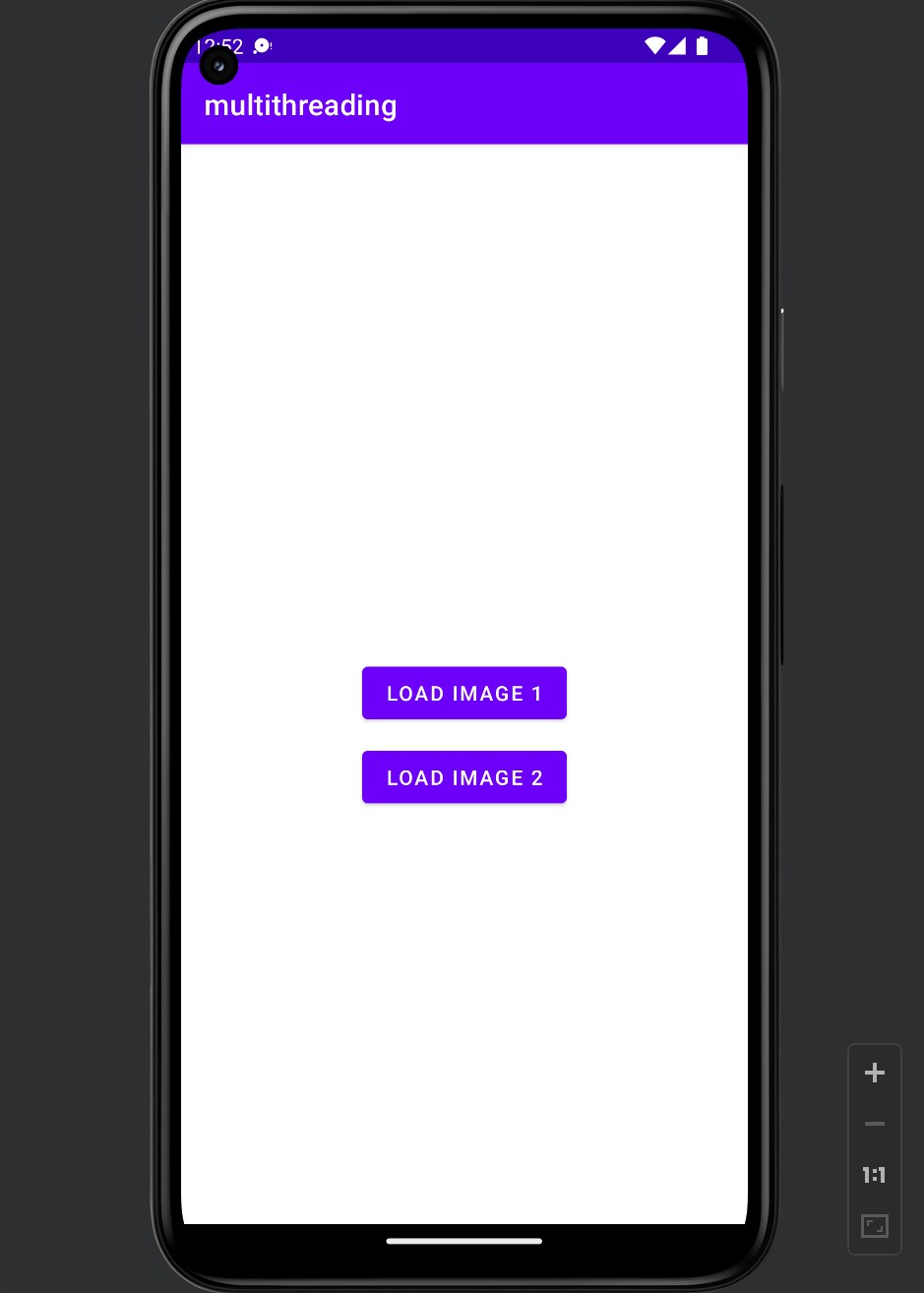
android:layout\_width="wrap\_content" android:layout\_height="wrap\_content" android:layout\_margin="5dp"

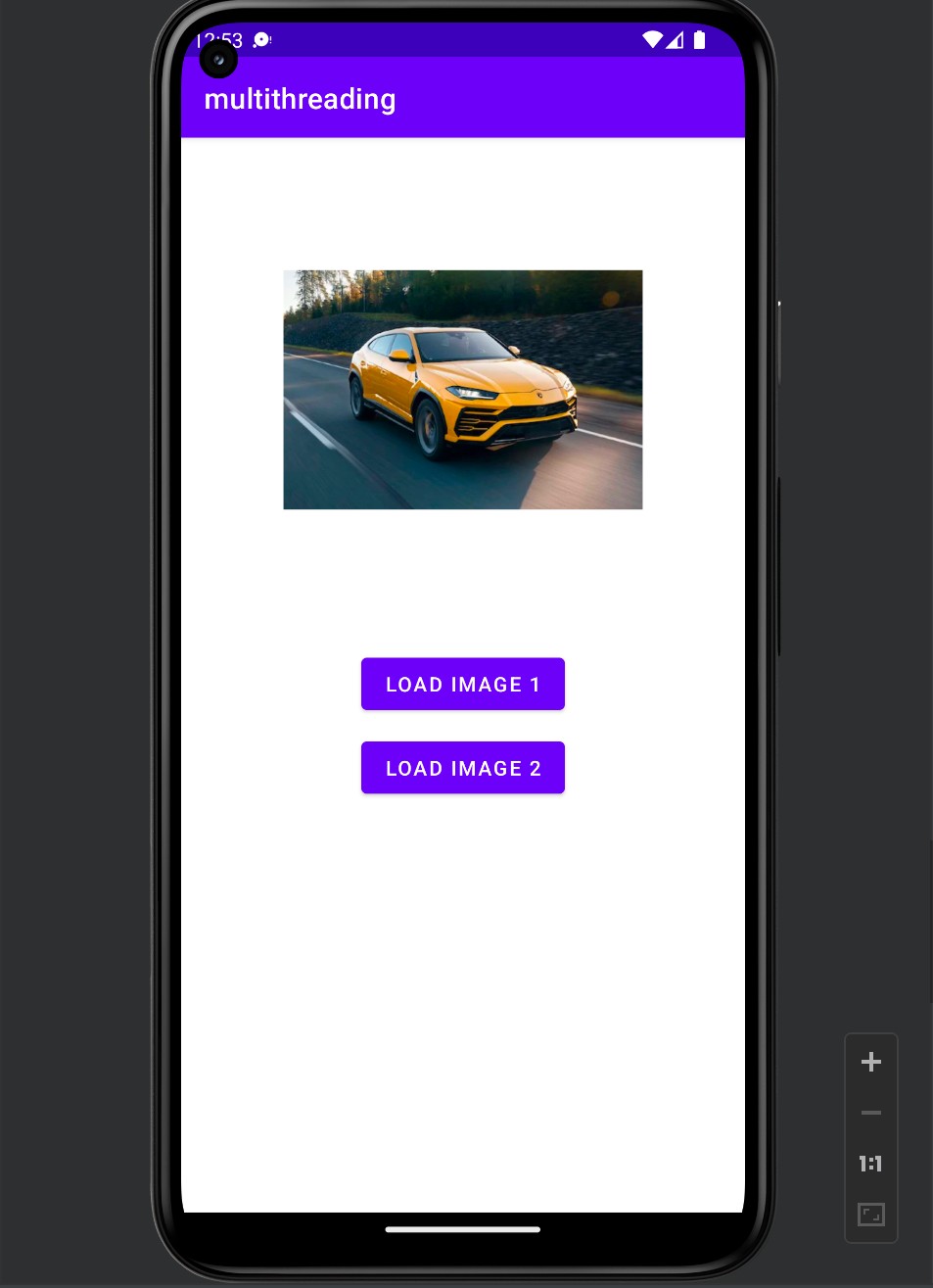
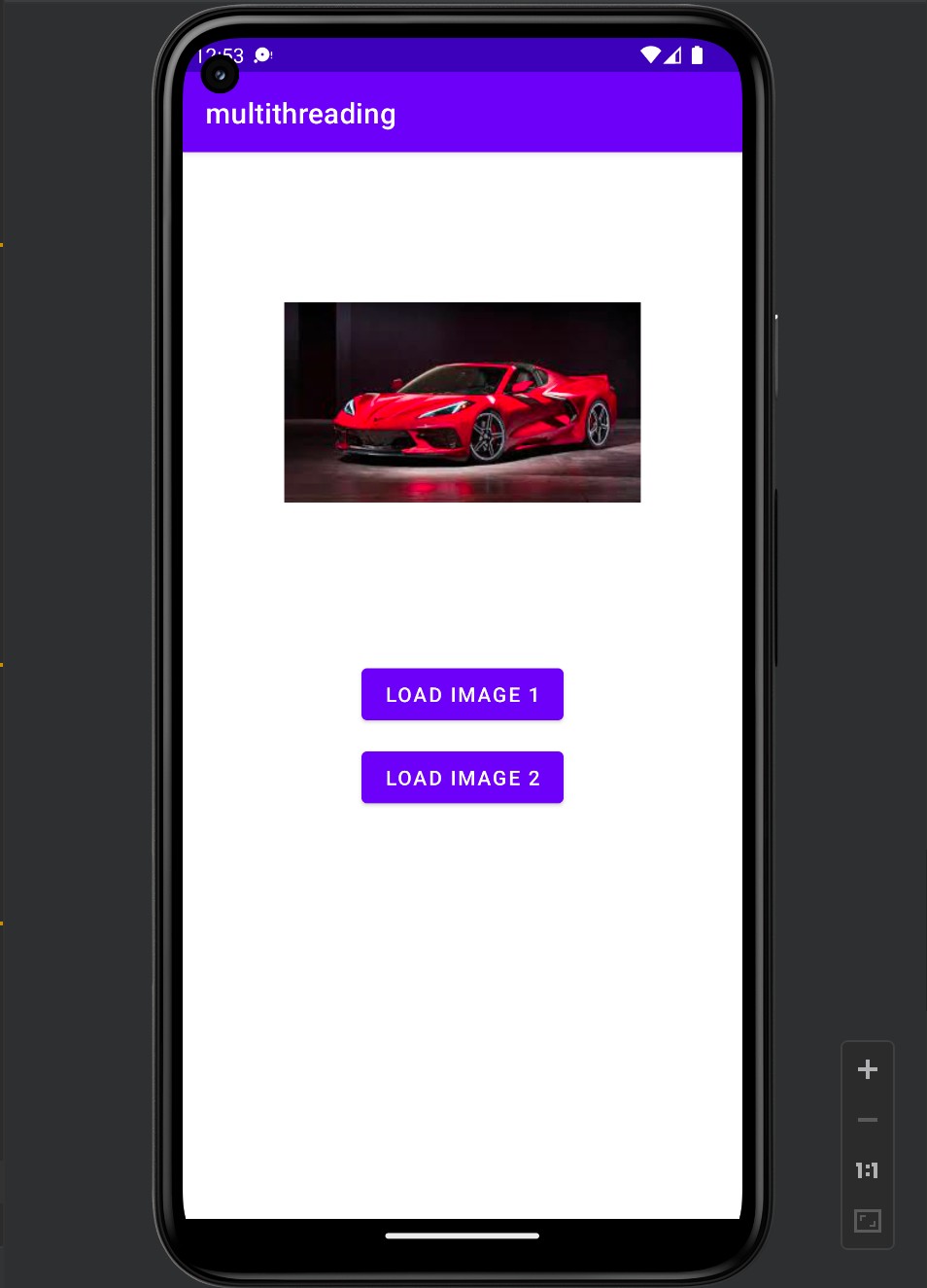
android:layout\_gravity="center"

android:text="Load image 2" />

</LinearLayout>

## Sample I/O:



## Result:

Thus, an Android Application that implements Multithreading has been successfully implemented.

**Develop a native application that uses GPS location information**

**Ex.No: 8**

Date: 01/11/2022

AIM:

To develop a native application that uses GPS location information. PROCEDURE:

* Install the following packages: geolocator & geocoding
* Import them using,

import ‘package: geocoding/geocoding.dart';

* import 'package:geolocator/geolocator.dart';
* Get the current location of the device, by creating an instance of Geolocator and calling

getCurrentPosition.

* Convert latitude and longitude values into addresses using placemarkFromCoordinates().

**CODE:**

import 'package:flutter/material.dart'; import ‘package:geocoding/geocoding.dart'; import 'package:geolocator/geolocator.dart';

class LocationPage extends StatefulWidget { @override

\_LocationPageState createState() => \_LocationPageState();

}

class \_LocationPageState extends State<LocationPage> { Position? \_currentPosition; String \_currentAddress = '';

@override

Widget build(BuildContext context) { return Scaffold( appBar: AppBar(

iconTheme: IconThemeData(

color: Colors.black, //change your color here

),

backgroundColor: Color(0xffef2e6c),

title: Text("Location",style:TextStyle(color:Colors.black)),

),

body: Center(

child: Column(

mainAxisAlignment: MainAxisAlignment.center, children: <Widget>[ Image.asset('assets/images/undraw\_Current\_location\_re\_j130.png'), TextButton(

style: ButtonStyle(backgroundColor: MaterialStateProperty.all(Color(0xffef2e6c))), child: Text("Get location",style:TextStyle(fontSize: 20,color:Colors.white)), onPressed: () {

Mohana R 195002073

\_getCurrentLocation(); },

),

Divider(color:Colors.transparent,thickness: 150), if (\_currentAddress != null) Text(

\_currentAddress,style: TextStyle(fontSize: 20), ),

if (\_currentPosition != null) Text( 'Latitude : ' + \_currentPosition!.latitude.toString(),style: TextStyle(fontSize: 20),

),

if (\_currentPosition != null) Text( 'Longitude : ' +

\_currentPosition!.longitude.toString(),style: TextStyle(fontSize: 20), ),

], ),

), );

}

\_getCurrentLocation() { Geolocator

.getCurrentPosition(desiredAccuracy: LocationAccuracy.best, forceAndroidLocationManag- er: true)

.then((Position position) { setState(() {

\_currentPosition = position;

\_getAddressFromLatLng(); });

}).catchError((e) { print(e);

}); }

\_getAddressFromLatLng() async { try {

List<Placemark> placemarks = await placemarkFromCoordinates( \_currentPosition!.latitude,

\_currentPosition!.longitude

);

Placemark place = placemarks[0]; setState(() {

\_currentAddress = "${place.locality}, ${place.postalCode}, ${place.country}";

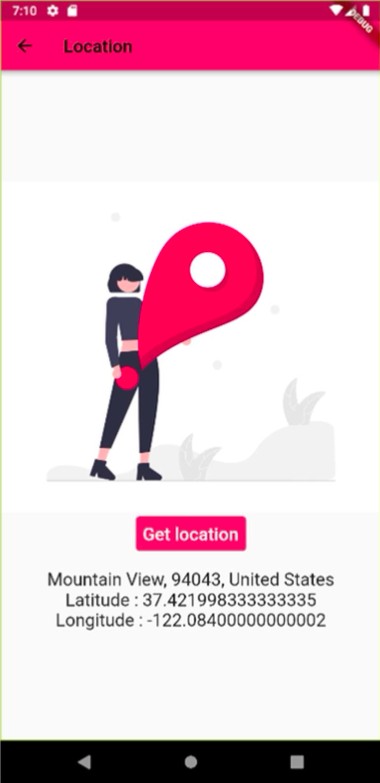
});

} catch (e) {

print(e); }

} }

## Output:



**RESULT:**

Thus, a native application that uses GPS location information has been devel- oped.

**SD Card**

**Ex.No: 9**

Date: 08/11/2022

**Aim:**

To develop an Android Application that writes data to the SD Card.

PROCEDURE:

* Install path\_provider package
* The path where is file is to be written is obtained using the getEter- nalStorageDirectory()

function.

* writeAsString(<String>) is used to write contents into a text file.
* readAsString() is used to read the contents of the file.

CODE:

import 'dart:async'; import 'dart:io';

import 'package:flutter/material.dart';

import 'package:path\_provider/path\_provider.dart';

class SDcard extends StatefulWidget { @override

\_AppState createState() => \_AppState();

}

class \_AppState extends State<SDcard> { String data=''; Future<String> get \_localPath async {

final directory = await getExternalStorageDirectory(); print(directory?.path); return directory!.path;

}

Future<File> get \_localFile async { final path = await \_localPath; return File('$path/ counter.txt');

}

Future<String> readContent() async { try { final file = await \_localFile;

// Read the file

String contents = await file.readAsString();

// Returning the contents of the file return contents; } catch (e) {

// If encountering an error, return return 'Error!'; }

}

Future<File> writeContent() async { final file = await \_localFile;

// Write the file

return file.writeAsString('One Point App');

}

@override

void initState() {

super.initState();

writeContent(); readContent().then((String value) { setState(() { data = value;

}); });

}

@override

Widget build(BuildContext context) { return Scaffold( appBar: AppBar(

title: Text("Write to SD Card", style: TextStyle(color: Colors.black)), leading: GestureDetec- tor(

child: Icon( Icons.arrow\_back\_ios, color: Colors.black, ), onTap: () { Navigator.pop(context); },

),

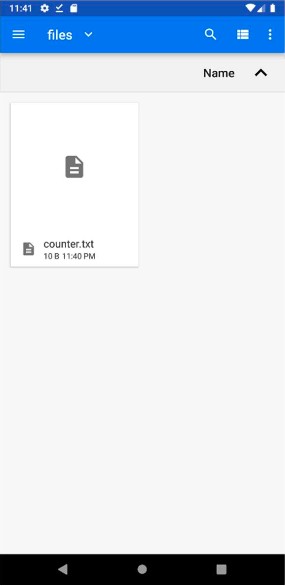
backgroundColor: Color(0xffef2e6c), ), body: Center( child: Text(

'Data read from a file: \n $data',style:TextStyle(fontSize: 40) ),

), );

} }

## Sample I/O:



**Result:**

Thus, an Android Application that writes data to the SD Card was successfully implemented.

**Alert**

**Ex.No: 10**

Date: 15/11/2022

**Aim:**

To develop an Android Application that creates an alert upon receiving a message.

**Software and Hardware Requirements:**

Software:

1. Android Studio
2. Java Hardware:
   1. Preferably 8GB+ RAM Laptop/Desktop

**Algorithm:**

1. Open Android Studio and then click on File -> New -> New project.
2. Then type the Application name as “alert″ and click Next.
3. Then select the Empty Activity and click Next.
4. In this application, a java class is created that makes use of BroadCastReceiver and SmsManager to get the sender number of the SMS along with the text that was sent.
5. These details are given as a toast message alert when the SMS is received by the device. This application uses the SMS permission to SEND, RECEIVE and READ SMS.

**Code: MainActivity.java:**

package com.example.alert;

import android.app.Notification;

import android.app.NotificationManager; import android.app.PendingIntent; import android.content.Intent;

import android.os.Bundle; import android.view.View; import android.widget.Button; import android.widget.EditText;

import androidx.appcompat.app.AppCompatActivity;

public class MainActivity extends AppCompatActivity{ Button notify;

EditText e; @Override

public void onCreate(Bundle savedInstanceState){ super.onCreate(savedInstanceState); setContentView(R.layout.activity\_main); notify= (Button) findViewById(R.id.button);

e= (EditText) findViewById(R.id.editText); notify.setOnClickListener(new View.OnClickListener() {

@Override

public void onClick(View v){

Intent intent = new Intent(MainActivity.this, SecondActivity.class);

PendingIntent pending = PendingIntent.getActivity(MainActivity.this, 0, intent, 0); Notification noti = new

Notification.Builder(MainActivity.this).setContentTitle("New Message").setContentText(e.getText().toString()).setSmallIcon(R.mipmap.ic\_launcher).setCo ntentIntent(pending).build();

NotificationManager manager = (NotificationManager) getSystemService(NOTIFICATION\_SERVICE);

noti.flags |= Notification.FLAG\_AUTO\_CANCEL; manager.notify(0, noti);

}

});

}

}

## Activity\_main.xml:

<?xml version="1.0" encoding="utf-8"?>

<LinearLayout xmlns:androi[d="http://schemas.android.com/apk/res/android"](http://schemas.android.com/apk/res/android) android:layout\_width="match\_parent" android:layout\_height="match\_parent"

android:layout\_margin="10dp" android:orientation="vertical">

<TextView android:layout\_width="wrap\_content" android:layout\_height="wrap\_content" android:text="Message" android:textSize="30sp" />

<EditText android:id="@+id/editText"

android:layout\_width="match\_parent" android:layout\_height="wrap\_content" android:singleLine="true" android:textSize="30sp" />

<Button

android:id="@+id/button" android:layout\_width="wrap\_content" android:layout\_height="wrap\_content" android:layout\_margin="30dp" android:layout\_gravity="center" android:text="Notify" android:textSize="30sp"/>

</LinearLayout>

## OUTPUT:

**RESULT:**

Thus Android Application that creates an alert upon receiving a message is developed and executed successfully

**Alarm**

**Ex.No: 11**

Date: 22/11/2022

**Aim:**

To develop an Android Application that creates an alert upon receiving a message.

**Software and Hardware Requirements:**

Software:

* 1. Android Studio
  2. Java Hardware:
     1. Preferably 8GB+ RAM Laptop/Desktop

**Algorithm:**

1. Open Android Studio and then click on File -> New -> New project.
2. Then type the Application name as “alert″ and click Next.
3. Then select the Empty Activity and click Next.
4. In the xml file, we have added two items ‘TimePicker’ and ‘ToggleButton’. TimePicker is used to capture the alarm time and ToggleButton is added to set the alarm on or off.
5. In MainActivity.java class the onToggleClicked( ) method is implemented in which the current hour and the minute is set using the calendar.
6. Create a new java class named “AlarmReceiver.java” at the same place where MainActivity.java class resides. In this class the onReceive() method is implemented. Here we have added vibration functionality and a default ringtone that starts to vibrate and ring when the alarm time is scheduled.
7. Go to the “values” folder first then choose the colors.xml file. In the colors.xml file, you can keep colors of your choice as many as you want to use in your app.
8. Go to the “values” folder first then choose the themes.xml file. In the theme.xml file, we have used “Theme.AppCompat.Light.DarkActionBar” which is a light theme with a dark ActionBar
9. Go to the “AndroidManifest.xml” file. A BroadcastReceiver is registered in AndroidManifest.xml by adding a receiver section after the application section is over.

**Code:**

**MainActivity.java:**

package com.example.alarmclock; import android.app.AlarmManager; import android.app.PendingIntent; import android.content.Intent; import android.os.Bundle;

import android.view.View;

import android.widget.TimePicker; import android.widget.Toast;

import android.widget.ToggleButton;

import androidx.appcompat.app.AppCompatActivity; import java.util.Calendar;

public class MainActivity extends AppCompatActivity { TimePicker alarmTimePicker;

PendingIntent pendingIntent; AlarmManager alarmManager;

@Override

protected void onCreate(Bundle savedInstanceState) { super.onCreate(savedInstanceState); setContentView(R.layout.activity\_main);

alarmTimePicker = (TimePicker) findViewById(R.id.timePicker); alarmManager = (AlarmManager) getSystemService(ALARM\_SERVICE);

}

// OnToggleClicked() method is implemented the time functionality public void OnToggleClicked(View view) {

long time;

if (((ToggleButton) view).isChecked()) {

Toast.makeText(MainActivity.this, "ALARM ON", Toast.LENGTH\_SHORT).show(); Calendar calendar = Calendar.getInstance();

// calender is called to get current time in hour and minute calendar.set(Calendar.HOUR\_OF\_DAY, alarmTimePicker.getCurrentHour()); calendar.set(Calendar.MINUTE, alarmTimePicker.getCurrentMinute());

// using intent i have class AlarmReceiver class which inherits

// BroadcastReceiver

Intent intent = new Intent(this, AlarmReceiver.class);

// we call broadcast using pendingIntent

pendingIntent = PendingIntent.getBroadcast(this, 0, intent, 0);

time = (calendar.getTimeInMillis() - (calendar.getTimeInMillis() % 60000)); if (System.currentTimeMillis() > time) {

// setting time as AM and PM if (calendar.AM\_PM == 0)

time = time + (1000 \* 60 \* 60 \* 12); else

time = time + (1000 \* 60 \* 60 \* 24);

}

// Alarm rings continuously until toggle button is turned off alarmManager.setRepeating(AlarmManager.RTC\_WAKEUP, time, 10000,

pendingIntent);

// alarmManager.set(AlarmManager.RTC\_WAKEUP, System.currentTimeMillis() + (time \* 1000), pendingIntent);

} else {

alarmManager.cancel(pendingIntent); Toast.makeText(MainActivity.this, "ALARM OFF",

Toast.LENGTH\_SHORT).show();

}

}

}

## Activity\_main.xml:

<?xml version="1.0" encoding="utf-8"?>

<LinearLayout xmlns:androi[d="http://schemas.android.com/apk/res/android"](http://schemas.android.com/apk/res/android) android:layout\_width="match\_parent" android:layout\_height="match\_parent" android:orientation="vertical">

<!--Added Time picker just to pick the alarm time-->

<!--gravity is aligned to center-->

<TimePicker android:id="@+id/timePicker"

android:layout\_width="wrap\_content" android:layout\_height="wrap\_content" android:layout\_gravity="center" />

<!--Added Toggle Button to set the alarm on or off-->

<!--ByDefault toggleButton is set to false-->

<ToggleButton android:id="@+id/toggleButton" android:layout\_width="wrap\_content" android:layout\_height="wrap\_content" android:layout\_gravity="center" android:layout\_margin="20dp" android:checked="false" android:onClick="OnToggleClicked" />

<!--"OnToggleClicked" method will be implemented in MainActivity.java -->

</LinearLayout>

## AlarmReceiver.java

package com.example.alarmclock;

import android.content.BroadcastReceiver; import android.content.Context;

import android.content.Intent; import android.media.Ringtone;

import android.media.RingtoneManager; import android.net.Uri;

import android.os.Build; import android.os.Vibrator; import android.widget.Toast;

import androidx.annotation.RequiresApi;

public class AlarmReceiver extends BroadcastReceiver { @RequiresApi(api = Build.VERSION\_CODES.Q) @Override

// implement onReceive() method

public void onReceive(Context context, Intent intent) {

// we will use vibrator first Vibrator vibrator = (Vibrator)

context.getSystemService(context.VIBRATOR\_SERVICE); vibrator.vibrate(4000);

Toast.makeText(context, "Alarm! Wake up! Wake up!", Toast.LENGTH\_LONG).show();

Uri alarmUri = RingtoneManager.getDefaultUri(RingtoneManager.TYPE\_ALARM); if (alarmUri == null) {

alarmUri = RingtoneManager.getDefaultUri(RingtoneManager.TYPE\_NOTIFICATION);

}

// setting default ringtone

Ringtone ringtone = RingtoneManager.getRingtone(context, alarmUri);

// play ringtone ringtone.play();

}

}

## Themes.xml:

<resources xmlns:tool[s="http://schemas.android.com/tools](http://schemas.android.com/tools)">

<!-- Base application theme. -->

<style name="Theme.Alarmclock" parent="Theme.MaterialComponents.DayNight.DarkActionBar">

<!-- Primary brand color. -->

<item name="colorPrimary">@color/purple\_500</item>

<item name="colorPrimaryVariant">@color/purple\_700</item>

<item name="colorOnPrimary">@color/white</item>

<!-- Secondary brand color. -->

<item name="colorSecondary">@color/teal\_200</item>

<item name="colorSecondaryVariant">@color/teal\_700</item>

<item name="colorOnSecondary">@color/black</item>

<!-- Status bar color. -->

<item name="android:statusBarColor" tools:targetApi="l">?attr/colorPrimaryVariant</ item>

<!-- Customize your theme here. -->

</style>

<style name="AppTheme" parent="Theme.AppCompat.Light.DarkActionBar">

<!-- Customize your theme here. -->

<item name="colorPrimary">@color/colorPrimary</item>

<item name="colorPrimaryDark">@color/colorPrimaryDark</item>

<item name="colorAccent">@color/colorAccent</item>

</style>

</resources>

## Androidmanifest.xml:

<?xml version="1.0" encoding="utf-8"?>

<manifest xmlns:androi[d="http://schemas.android.com/apk/res/android"](http://schemas.android.com/apk/res/android) package="com.example.alarmclock">

<application android:allowBackup="true" android:icon="@mipmap/ic\_launcher" android:label="@string/app\_name"

android:roundIcon="@mipmap/ic\_launcher\_round" android:supportsRtl="true" android:theme="@style/Theme.alarmclock">

<activity android:name=".MainActivity">

<intent-filter>

<action android:name="android.intent.action.MAIN" />

<category android:name="android.intent.category.LAUNCHER" />

</intent-filter>

</activity>

</application>

<uses-permission android:name="android.permission.VIBRATE" />

</manifest> Colors.xml:

<?xml version="1.0" encoding="utf-8"?>

<resources>

<color name="purple\_200">#FFBB86FC</color>

<color name="purple\_500">#FF6200EE</color>

<color name="purple\_700">#FF3700B3</color>

<color name="teal\_200">#FF03DAC5</color>

<color name="teal\_700">#FF018786</color>

<color name="black">#FF000000</color>

<color name="white">#FFFFFFFF</color>

<color name="colorPrimary">#0F9D58</color>

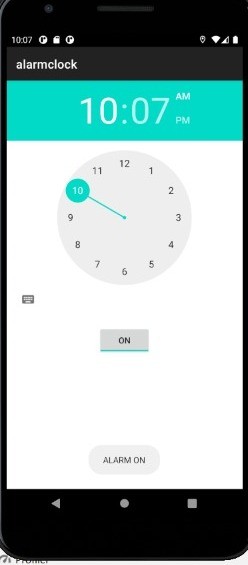
<color name="colorPrimaryDark">#0F4C2E</color>

<color name="colorAccent">#9D0F9B</color>

</resources>

## OUTPUT:





**RESULT:**

Thus Android Application that creates an Alarm Clock is developed and executed successfully.

**Ex. No: 12**

Date: 27/11/2022

**Game - Tic Tac Toe**

**Aim:**

To develop a simple gaming application with multimedia support

**Software and Hardware Requirements:**

Software:

* 1. Android Studio
  2. Java Hardware:

1. Preferably 8GB+ RAM Laptop/Desktop

**Algorithm:**

1. Open Android Studio and create a new project.
2. Select Empty Activity.
3. Create a function onClick() to add user input as ‘x’ or ‘o’ based on the player.
4. Create a function checkWin() to check if any of the three columns/ rows/ diagonals are having the same value.
5. Create separate functions for player 1 and 2 wins .
6. Within player wins ,add audio using Mediaplayer.
7. Use functions like .start(), . create() and .release() to play and pause the audio.
8. Finally, run the code

**Code: MainActivity.java:**

package com.example.tictactoe;

import androidx.appcompat.app.AppCompatActivity; import android.media.MediaPlayer;

import android.os.Bundle; import android.view.View; import android.widget.Button;

import android.widget.ImageSwitcher; import android.widget.ImageView; import android.widget.TextView; import android.widget.Toast;

public class MainActivity extends AppCompatActivity implements View.OnClickListener { private Button[][] buttons = new Button[3][3];

private boolean player1Turn = true; private int roundCount;

private int player1Points; private int player2Points;

private TextView textViewPlayer1; private TextView textViewPlayer2; MediaPlayer winner;

@Override

protected void onCreate(Bundle savedInstanceState) { super.onCreate(savedInstanceState); setContentView(R.layout.activity\_main);

textViewPlayer1 = findViewById(R.id.text\_view\_p1); textViewPlayer2 = findViewById(R.id.text\_view\_p2);

for (int i = 0; i < 3; i++) { for (int j = 0; j < 3; j++) {

String buttonID = "button\_" + i + j;

int resID = getResources().getIdentifier(buttonID, "id", getPackageName()); buttons[i][j] = findViewById(resID);

buttons[i][j].setOnClickListener(this);

}

}

Button buttonReset = findViewById(R.id.button\_reset);

buttonReset.setOnClickListener(new View.OnClickListener() { @Override

public void onClick(View v) { resetGame();

}

});

}

@Override

public void onClick(View v) {

if (!((Button) v).getText().toString().equals("")) { return;

}

if (player1Turn) {

// textViewPlayer1.setImageResource(R.drawable.yellow); ((Button) v).setText("X");

} else {

((Button) v).setText("O");

//textViewPlayer2.setImageResource(R.drawable.yellow);

}

roundCount++;

if (checkForWin()) { if (player1Turn) {

player1Wins();

} else {

player2Wins();

}

} else if (roundCount == 9) { draw();

} else {

player1Turn = !player1Turn;

}

}

private boolean checkForWin() { String[][] field = new String[3][3];

for (int i = 0; i < 3; i++) { for (int j = 0; j < 3; j++) {

field[i][j] = buttons[i][j].getText().toString();

}

}

for (int i = 0; i < 3; i++) {

if (field[i][0].equals(field[i][1])

&& field[i][0].equals(field[i][2])

&& !field[i][0].equals("")) { return true;

}

}

for (int i = 0; i < 3; i++) {

if (field[0][i].equals(field[1][i])

&& field[0][i].equals(field[2][i])

&& !field[0][i].equals("")) { return true;

}

}

if (field[0][0].equals(field[1][1])

&& field[0][0].equals(field[2][2])

&& !field[0][0].equals("")) { return true;

}

if (field[0][2].equals(field[1][1])

&& field[0][2].equals(field[2][0])

&& !field[0][2].equals("")) { return true;

}

return false;

}

private void player1Wins() { player1Points++;

winner = MediaPlayer.create(MainActivity.this,R.raw.win); winner.start();

Toast.makeText(this, "Player 1 wins!", Toast.LENGTH\_SHORT).show(); updatePointsText();

//winner.release(); resetBoard();

}

private void player2Wins() { player2Points++;

winner = MediaPlayer.create(MainActivity.this,R.raw.win); winner.start();

Toast.makeText(this, "Player 2 wins!", Toast.LENGTH\_SHORT).show(); updatePointsText();

resetBoard();

}

private void draw() {

Toast.makeText(this, "Draw!", Toast.LENGTH\_SHORT).show(); resetBoard();

}

private void updatePointsText() { textViewPlayer1.setText("Player 1: " + player1Points); textViewPlayer2.setText("Player 2: " + player2Points);

}

private void resetBoard() { for (int i = 0; i < 3; i++) {

for (int j = 0; j < 3; j++) { buttons[i][j].setText("");

}

}

roundCount = 0; player1Turn = true;

}

private void resetGame() { player1Points = 0;

player2Points = 0; updatePointsText(); resetBoard();

}

@Override

protected void onSaveInstanceState(Bundle outState) { super.onSaveInstanceState(outState);

outState.putInt("roundCount", roundCount); outState.putInt("player1Points", player1Points); outState.putInt("player2Points", player2Points); outState.putBoolean("player1Turn", player1Turn);

}

@Override

protected void onRestoreInstanceState(Bundle savedInstanceState) { super.onRestoreInstanceState(savedInstanceState);

roundCount = savedInstanceState.getInt("roundCount"); player1Points = savedInstanceState.getInt("player1Points"); player2Points = savedInstanceState.getInt("player2Points"); player1Turn = savedInstanceState.getBoolean("player1Turn");

}

}

activity\_main.xml

<?xml version="1.0" encoding="utf-8"?>

<LinearLayout xmlns:androi[d="http://schemas.android.com/apk/res/android"](http://schemas.android.com/apk/res/android) xmlns:app="[http://schemas.android.com/apk/res-auto"](http://schemas.android.com/apk/res-auto)

xmlns:tool[s="http://schemas.android.com/tools](http://schemas.android.com/tools)" android:layout\_width="match\_parent" android:layout\_height="match\_parent" android:orientation="vertical" tools:context="com.example.tictactoe.MainActivity">

<RelativeLayout android:layout\_width="match\_parent" android:layout\_height="wrap\_content">

<TextView android:id="@+id/text\_view\_p1" android:layout\_width="fill\_parent" android:layout\_height="wrap\_content" android:freezesText="true" android:text="Player 1: 0" android:textSize="30sp" />

<TextView android:id="@+id/text\_view\_p2" android:layout\_width="fill\_parent" android:layout\_height="wrap\_content" android:freezesText="true" android:text="Player 1: 0"

android:layout\_below="@+id/text\_view\_p1" android:textSize="30sp" />

<Button

android:id="@+id/button\_reset" android:layout\_width="wrap\_content" android:layout\_height="wrap\_content" android:layout\_alignParentEnd="true" android:layout\_centerVertical="true" android:layout\_marginEnd="33dp" android:text="reset" android:layout\_alignParentRight="true" android:layout\_marginRight="33dp" />

</RelativeLayout>

<LinearLayout android:layout\_width="match\_parent" android:layout\_height="0dp" android:layout\_weight="1">

<Button

android:id="@+id/button\_00" android:layout\_width="0dp" android:layout\_height="match\_parent" android:layout\_weight="1" android:freezesText="true" android:textSize="60sp" />

<Button

android:id="@+id/button\_01" android:layout\_width="0dp" android:layout\_height="match\_parent" android:layout\_weight="1" android:freezesText="true" android:textSize="60sp" />

<Button

android:id="@+id/button\_02" android:layout\_width="0dp" android:layout\_height="match\_parent" android:layout\_weight="1" android:freezesText="true" android:textSize="60sp" />

</LinearLayout>

<LinearLayout android:layout\_width="match\_parent" android:layout\_height="0dp" android:layout\_weight="1">

<Button

android:id="@+id/button\_10" android:layout\_width="0dp" android:layout\_height="match\_parent" android:layout\_weight="1" android:freezesText="true" android:textSize="60sp" />

<Button

android:id="@+id/button\_11" android:layout\_width="0dp" android:layout\_height="match\_parent" android:layout\_weight="1" android:freezesText="true" android:textSize="60sp" />

<Button

android:id="@+id/button\_12" android:layout\_width="0dp" android:layout\_height="match\_parent" android:layout\_weight="1" android:freezesText="true" android:textSize="60sp" />

</LinearLayout>

<LinearLayout android:layout\_width="match\_parent" android:layout\_height="0dp" android:layout\_weight="1">

<Button

android:id="@+id/button\_20" android:layout\_width="0dp" android:layout\_height="match\_parent" android:layout\_weight="1" android:freezesText="true" android:textSize="60sp" />

<Button

android:id="@+id/button\_21" android:layout\_width="0dp" android:layout\_height="match\_parent" android:layout\_weight="1" android:freezesText="true" android:textSize="60sp" />

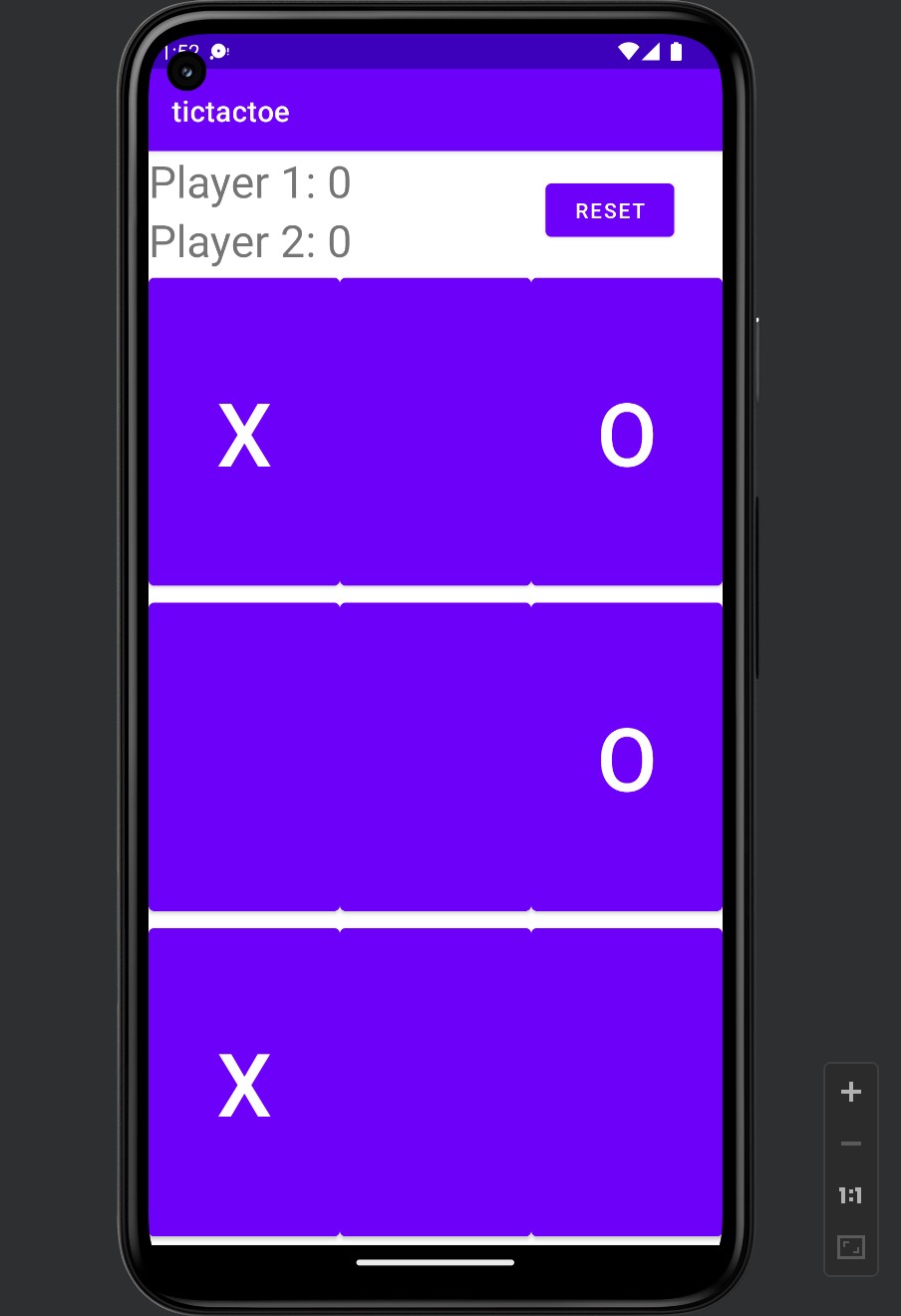
<Button

android:id="@+id/button\_22" android:layout\_width="0dp" android:layout\_height="match\_parent" android:layout\_weight="1" android:freezesText="true" android:textSize="60sp" />

</LinearLayout>

</LinearLayout>

## SAMPLE I/O :



**RESULT:**

Thus a simple gaming application with multimedia support has been built and executed successfully.

**CONNECTIVITY VIA SOAP OR REST**

**Ex.No: 13**

Date: 29/11/2022 AIM:

To a mobile application for data handling and connectivity via SOAP or REST to backend services potentially hosted in a cloud environment.

PROCEDURE:

* Import,HTTP.dart
* dart: convert
* Specify the URL of the API within “Uri.parse(<>)”
* HTTP.get() is used to fetch URL contents.

CODE:

**quotes.dart**

// To parse this JSON data, do

//

// final quotes = quotesFromJson(jsonString);

import 'dart:convert';

Quotes quotesFromJson(String str) => Quotes.fromJson(json.decode(str)); String quotesToJ- son(Quotes data) => json.encode(data.toJson());

class Quotes { Quotes({ this.id,

this.tags, this.content = '', this.author = '', this.authorSlug, this.length, this.dateAdded, this.-

dateModified,

});

String? id;

List<String>? tags; String content; String author;

String? authorSlug; int? length;

DateTime? dateAdded; DateTime? dateModified;

factory Quotes.fromJson(Map<String, dynamic> json) => Quotes(

id: json["\_id"],

tags: List<String>.from(json["tags"].map((x) => x)), content: json["content"], author: json["author"],

authorSlug: json["authorSlug"], length: json["length"],

dateAdded: DateTime.parse(json["dateAdded"]), dateModified: DateTime.parse(json["date- Modified"]),

);

Map<String, dynamic> toJson() => { "\_id": id,

"tags": List<dynamic>.from(tags!.map((x) => x)), "content": content,

"author": author, "authorSlug": authorSlug, "length": length, "dateAdded":

"${dateAdded!.year.toString().padLeft(4, '0')}-${dateAdded!.month.toString().padLeft(2,

'0')}-${dateAdded!.day.toString().padLeft(2, '0')}", "dateModified": "${dateModified!.year.- toString().padLeft(4, '0')}-

${dateModified!.month.toString().padLeft(2, '0')}-${dateModified!.day.toString().padLeft(2, '0')}",

}; }

**api.dart**

import 'dart:convert';

import 'package:http/http.dart' as http; import 'quotes.dart';

class Api {

static Future<Quotes?> getQuotes() async {

Uri url = Uri.parse[('http://api.quotable.io/random');](http://api.quotable.io/random%27)%3B) http.Response response = await http.get(url);

if (response.statusCode == 200) { print("success");

return Quotes.fromJson(jsonDecode(response.body));

} else {

print("error in getting data");

} }

}

**quotes\_page.dart**

import 'dart:convert';

import 'package:flutter/material.dart'; import 'package:http/http.dart' as http; import 'quotes.- dart';

import 'api.dart';

class QuotesScreen extends StatefulWidget { QuotesScreen({Key? key}) : super(key: key); @override

State<QuotesScreen> createState() => \_QuotesScreenState(); }

class \_QuotesScreenState extends State<QuotesScreen> { var size, height, width; Quotes? data;

@override

Widget build(BuildContext context) { size = MediaQuery.of(context).size; height = size.height;

width = size.width;

return Scaffold( appBar: AppBar(

backgroundColor: Color(0xffef2e6c), title: Text("Quotations"), actions: [

IconButton( icon: Icon( Icons.refresh\_outlined, ),

iconSize: 30, onPressed: () { print("icon refresh"); getQuotes(); },

), ],

),

body: RefreshIndicator(

onRefresh: getQuotes, child: ListView( children: [ Padding(

padding: const EdgeInsets.all(18.0), child: Text(

"Pull to Refresh", textAlign: TextAlign.center, style: TextStyle( fontSize: 15,

), ),

),

SizedBox(height: 20), Image.asset('assets/images/undraw\_Bibliophile\_re\_xarc.png'), Sized- Box(height: 20),

Container(

padding: EdgeInsets.symmetric( horizontal: 10,

),

width: width / 2, child: Card(

margin: EdgeInsets.only(top: 20), color: Color(0XFFeeeeee), shape: RoundedRectangleBor- der(

borderRadius: BorderRadius.circular(10.0), ), elevation: 10, child: Padding(

padding: EdgeInsets.symmetric(horizontal: 10, vertical: 20), child: Column( mainAxisAlignment: MainAxisAlignment.center, children: [

Text(

'${data?.content ?? "Don't talk about what you have done or what you are

going to do."}',

textAlign: TextAlign.justify,

style: TextStyle( fontSize: 20,

fontStyle: FontStyle.italic,

), ),

SizedBox(height: 22), Align(

alignment: Alignment.bottomRight, child: Text(

data?.author ?? "Thomas Jefferson", textAlign: TextAlign.justify, style: TextStyle(

fontSize: 17,

fontWeight: FontWeight.bold, ),

)) ],

), ),

), )

], ),

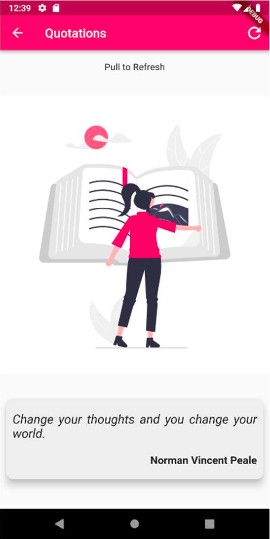
), );

}

Future<Null> getQuotes() async { data = await Api.getQuotes(); setState(() {});

} }

**OUTPUT:**



**RESULT:**

Hence, a mobile application for data handling and connectivity via SOAP or REST to back- end services potentially hosted in a cloud environment.

# GEO-POSITIONING, ACCELEROMETER AND RICH GESTURE-BASED UI

**EX.NO:14**

Date: 03/12/2022

AIM:

To write a mobile application that will take advantage of underlying phone functionality including GEO positioning, accelerometer, and rich gesture-based UI handling.

PROCEDURE:

Geo-positioning:

* + Install the following packages: geolocator & geocoding
  + Import them using,

import 'package:geocoding/geocoding.dart';

* + import ‘package: geolocator/geolocator.dart';
  + Get the current location of the device, by creating an instance of Geolocator and calling

getCurrentPosition.

* + Convert latitude and longitude values into address using placemarkFromCoordinates(). Accelerometer:
  + Install the sensors package.
  + Import it using, ‘import 'package:sensors/sensors.dart';’
  + Accelerometer readings tell if the device is moving in a particular direction.

Gesture-based UI:

* + In the onTap() property of the GestureDetector(), pass the function to be performed.
  + In this case, it reverses the boolean value isLightsOn.
  + This is used to switch the theme of the screen as dark or light.
  + The child property of GestureDetector() is used to specify icon, on clicking which the

action is to be performed.

Geo-positioning:

CODE:

import 'package:flutter/material.dart'; import 'package:geocoding/geocoding.dart'; import 'package:geolocator/geolocator.dart';

class LocationPage extends StatefulWidget { @override

\_LocationPageState createState() => \_LocationPageState();

}

class \_LocationPageState extends State<LocationPage> { Position? \_currentPosition; String \_currentAddress = '';

@override

Widget build(BuildContext context) {

return Scaffold( appBar: AppBar( iconTheme: IconThemeData(

color: Colors.black, //change your color here

),

backgroundColor: Color(0xffef2e6c),

title: Text("Location",style:TextStyle(color:Colors.black)),

),

body: Center(

child: Column(

mainAxisAlignment: MainAxisAlignment.center, children: <Widget>[ Image.asset('assets/images/undraw\_Current\_location\_re\_j130.png'), TextButton(

style: ButtonStyle(backgroundColor: MaterialStateProperty.all(Color(0xffef2e6c))), child: Text("Get location",style:TextStyle(fontSize: 20,color:Colors.white)), onPressed: () {

\_getCurrentLocation(); },

),

Divider(color:Colors.transparent,thickness: 150), if (\_currentAddress != null) Text(

\_currentAddress,style: TextStyle(fontSize: 20), ),

if (\_currentPosition != null) Text( 'Latitude : ' + \_currentPosition!.latitude.toString(),style: TextStyle(fontSize: 20),

),

if (\_currentPosition != null) Text( 'Longitude : ' +

\_currentPosition!.longitude.toString(),style: TextStyle(fontSize: 20), ),

], ),

), );

}

\_getCurrentLocation() { Geolocator

.getCurrentPosition(desiredAccuracy: LocationAccuracy.best, forceAndroidLocationManag- er: true)

.then((Position position) { setState(() {

\_currentPosition = position;

\_getAddressFromLatLng(); });

}).catchError((e) { print(e);

});

}

\_getAddressFromLatLng() async { try {

List<Placemark> placemarks = await placemarkFromCoordinates( \_currentPosition!.latitude,

\_currentPosition!.longitude

);

Placemark place = placemarks[0]; setState(() {

\_currentAddress = "${place.locality}, ${place.postalCode}, ${place.country}";

});

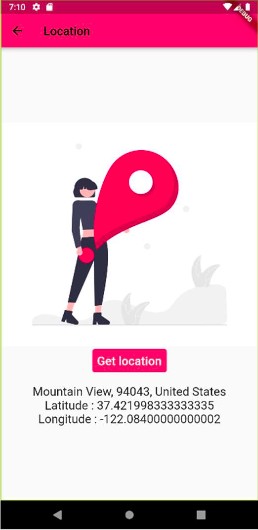
} catch (e) {

print(e); }

}

}

OUTPUT:



**Accelerometer:**

CODE:

import 'dart:async';

import 'package:flutter/material.dart'; import 'package:sensors/sensors.dart';

}); }

}

pauseTimer() {

// stop the timer and pause the accelerometer stream timer?.cancel(); accel?.pause();

// set the success color and reset the count

setState(() { count = 0;

color = Colors.green;

}); }

@override

void dispose() {

timer?.cancel(); accel?.cancel(); super.dispose();

}

@override

Widget build(BuildContext context) {

// get the width and height of the screen

width = MediaQuery.of(context).size.width; height = MediaQuery.of(context).size.height; return Scaffold( appBar: AppBar(

iconTheme: IconThemeData(

color: Colors.black, //change your color here

),

title: Text(widget.title,style:TextStyle(color:Colors.black)), backgroundColor : Color(0xffe- f2e6c),

),

body: Column( children: [ Padding(

padding: const EdgeInsets.all(8.0),

child: Text('Keep the circle in the center for 1 second',textAlign: TextAlign.center,style: TextStyle(fontSize:25)),

), Stack( children: [

// This empty container is given a width and height to set the size of the stack

Container(

height: height! / 2, width: width,

),

// Create the outer target circle wrapped in a Position Positioned(

// positioned 50 from the top of the stack

// and centered horizontally, left = (ScreenWidth - Container width) / 2 top: 50, left: (width! - 250) / 2,

child: Container(

height: 250,

width: 250,

decoration: BoxDecoration(

border: Border.all(color: Colors.red, width: 5.0),

borderRadius: BorderRadius.circular(125), ),

), ),

// This is the colored circle that will be moved by the accelerometer // the top and left are variables that will be set

Positioned(

top: top,

left: left ?? (width! - 100) / 2,

// the container has a color and is wrappeed in a ClipOval to make it round child: ClipOval( child: Container( width: 100, height: 100, color: color,

), ),

),

// inner target circle wrapped in a Position Positioned(

top: 125,

left: (width! - 100) / 2, child: Container(

height: 100,

width: 100,

decoration: BoxDecoration(

border: Border.all(color: Colors.green, width: 2.0), borderRadius: BorderRadius.circular(50), ),

), ),

], ),

Text('x: ${(event?.x ?? 0).toStringAsFixed(3)}',style:TextStyle(fontSize: 20)), Text('y: $

{(event?.y ?? 0).toStringAsFixed(3)}',style:TextStyle(fontSize: 20)), Padding( padding: EdgeInsets.symmetric(horizontal: 16.0, vertical: 30.0),

child: TextButton(

style: ButtonStyle(backgroundColor: MaterialStateProperty.all(Color(0xffef2e6c))), on- Pressed: startTimer,

child: Text('Begin.!!',style: TextStyle(fontSize: 30.0,color:Colors.white),),

// color: Theme.of(context).primaryColor,

// textColor: Colors.white,

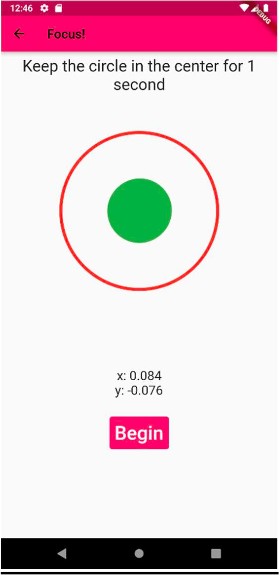
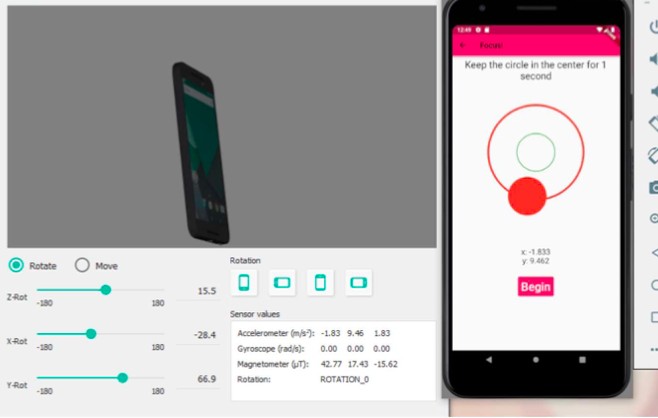
), )

], ),

); }

}

OUTPUT:



**Gesture-based UI:**

**CODE:**

import 'package:flutter/material.dart';

import 'package:google\_fonts/google\_fonts.dart';

class AboutPage extends StatefulWidget { @override

\_AboutPageState createState() => \_AboutPageState();

}

class \_AboutPageState extends State<AboutPage> { bool \_lightIsOn = false; @override

void dispose() { super.dispose(); }

@override

void initState() { super.initState(); }

@override

Widget build(BuildContext context) {

return MaterialApp(

theme: \_lightIsOn ? ThemeData.dark() : ThemeData.light(), home: Scaffold( appBar: AppBar(

title: Text('About', style: TextStyle(color: Colors.black)), backgroundColor: Color(0xffef2e6c), ),

body: Column(children: <Widget>[ Container(

margin: EdgeInsets.all(20), height: 200,

width: 350,

child: Image.asset('assets/images/logo.png'),

),

Divider(color:Colors.black,thickness: 2,), Container(

// alignment: FractionalOffset.center, child: Column(

// mainAxisAlignment: MainAxisAlignment.center, children: <Widget>[ GestureDetector(

onTap: () { setState(() {

// Toggle light when tapped.

\_lightIsOn = !\_lightIsOn; });

},

child: Container(

margin: EdgeInsets.fromLTRB(350, 10, 3, 6), width : 50, height:50,

padding: const EdgeInsets.all(8),

// Change button text when light changes state. decoration: BoxDecoration( shape : BoxShape.circle,

color: Color(0xffef2e6c), ),

child: Icon(

\_lightIsOn ? Icons.light\_mode\_outlined : Icons.dark\_mode\_outlined, size: 30),

), ),

], ),

),

Text('In publishing and graphic design, '

'Lorem ipsum is a placeholder text commonly used to demonstrate ' 'the visual form of a doc- ument or a typeface without relying on ' 'meaningful content. Lorem ipsum may be used as a placeholder ' 'before final copy is available.',

textAlign: TextAlign.center, softWrap: true,

style: GoogleFonts.notoSerif(textStyle: TextStyle( color: \_lightIsOn ? Colors.white : Colors.black,fontSize: 20),) ),

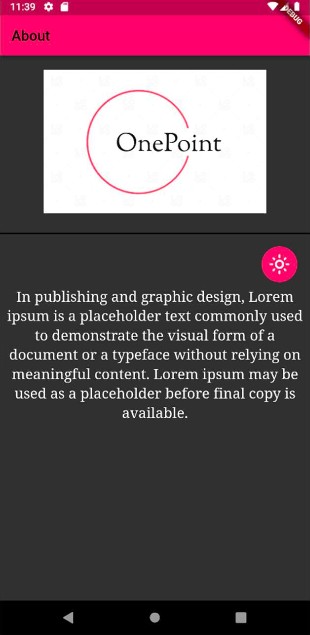
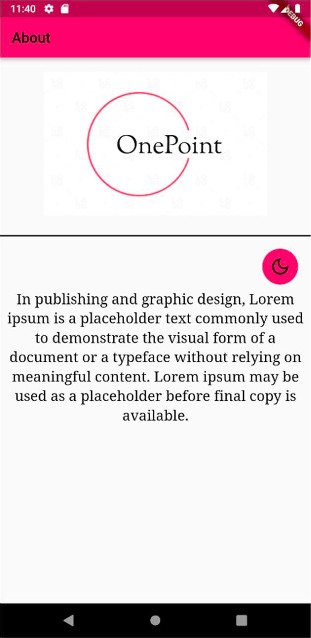
]

) )

); }}

OUTPUT:

Dark modeLight Mode



**RESULT:**

Thus, GEO positioning, accelerometer, and rich gesture-based UI handling have been implemented using Flutter.

**SOCIAL MEDIA INTEGRATION**

**EX.NO:15**

Date: 06/12/2022

**AIM:**

To write an application for integrating mobile applications in the market, including

social networking software integration with Google.

**PROCEDURE:**

-  Download the following packages using flutter pub add. o firebase\_auth o firebase\_core o google\_sign\_in

-  In the firebase console, enable Google as a provider under Authentication-> Sign In method.

-  Get SHA key, by using the command gradlew signingReport at the android directory of the flutter application.

-  Add SHA-1 fingerprint to the application.

-  Now, get Google user credential using the await GoogleSignIn().signIn();

-  Obtain the auth details from the request.

-  Obtain the auth details from the request

**CODE:**

authentication.dart

page74image55068736

import 'package:firebase\_auth/firebase\_auth.dart'; import 'package:google\_sign\_in/google\_sign\_in.dart';

class AuthenticationHelper {  
final FirebaseAuth \_auth = FirebaseAuth.instance;

get user => \_auth.currentUser;

Future<String?> signInWithGoogle() async {  
final GoogleSignInAccount? googleUser = await GoogleSignIn().signIn();

final GoogleSignInAuthentication? googleAuth = await googleUser?.authentication;

final credential = GoogleAuthProvider.credential( accessToken: googleAuth?.accessToken, idToken: googleAuth?.idToken, );

await FirebaseAuth.instance.signInWithCredential(credential);

return null;

}

Future<UserCredential> signInWithFacebook() async {

// Trigger the sign-in flow  
final LoginResult loginResult = await FacebookAuth.instance.login();

// Create a credential from the access token

final OAuthCredential facebookAuthCredential = FacebookAuthProvider.credential(loginResult.accessToken.token);

// Once signed in, return the UserCredential

return FirebaseAuth.instance.signInWithCredential(facebookAuthCredential); }

//SIGN UP METHOD

Future<String?> signUp({required String email, required String password}) async { try {

await \_auth.createUserWithEmailAndPassword( email: email,  
password: password,

);

return null;  
} on FirebaseAuthException catch (e) {

return e.message; }

}

//SIGN IN METHOD

Future<String?> signIn({required String email, required String password}) async { try {

await \_auth.signInWithEmailAndPassword(email: email, password: password);

return null;  
} on FirebaseAuthException catch (e) {

return e.message; }

}

//SIGN OUT METHOD

Future<void> signOut() async { await \_auth.signOut();

print('signout'); }

}

login.dart

import 'package:flutter/material.dart'; import './authentication.dart';  
import './home.dart';  
import './signup.dart';

class Login extends StatelessWidget { @override  
Widget build(BuildContext context) {

return Scaffold( body: ListView(

padding: EdgeInsets.all(8.0), children: <Widget>[

SizedBox(height: 80),

// logo

Column( children: [

Image.asset('assets/images/logo.png'), SizedBox(height: 50),  
Text(

'Welcome back!',

style: TextStyle(fontSize: 24), ),

], ),

SizedBox( height: 50,

),

Padding(  
padding: const EdgeInsets.all(16.0), child: LoginForm(),

),

SizedBox(height: 20), Row(

children: <Widget>[ SizedBox(width: 30), Text('New here ? ',

style: TextStyle(fontWeight: FontWeight.bold, fontSize: 20)), GestureDetector(

onTap: () {  
Navigator.pushReplacement(context,MaterialPageRoute(builder: (context) =>

Signup())); },

child: Text('Get Registered Now..',  
style: TextStyle(fontSize: 20, color: Color(0xffef2e6c))),

) ],

), Row(

children: <Widget>[ SizedBox(width: 30), GestureDetector(

onTap: () { AuthenticationHelper()

.signInWithGoogle()

.then((result) {  
if (result == null) {

Navigator.pushReplacement(context, MaterialPageRoute(builder: (context) => MyApp()));

} else { ScaffoldMessenger.of(context).showSnackBar(SnackBar(

content: Text(  
result,  
style: TextStyle(fontSize: 16),

), ));

} });

},  
child: Text('Sign in with Google',

style: TextStyle(fontSize: 20, color: Color(0xffef2e6c))), )

], ),

], ),

); }

}

class LoginForm extends StatefulWidget { LoginForm({Key? key}) : super(key: key);

@override

\_LoginFormState createState() => \_LoginFormState(); }

class \_LoginFormState extends State<LoginForm> { final \_formKey = GlobalKey<FormState>();

String? email; String? password;

bool \_obscureText = true;

@override  
Widget build(BuildContext context) {

return Form( key: \_formKey, child: Column(

mainAxisAlignment: MainAxisAlignment.spaceAround, children: <Widget>[

// email

TextFormField(

// initialValue: 'Input text',

decoration: InputDecoration(  
prefixIcon: Icon(Icons.email\_outlined,color:Colors.black), labelText: 'Email',  
labelStyle: TextStyle(

color: Color(0xffef2e6c), ),

enabledBorder: OutlineInputBorder( borderRadius: BorderRadius.all(

const Radius.circular(100.0), ),

),  
focusedBorder: OutlineInputBorder(

borderRadius: BorderRadius.all( const Radius.circular(100.0),

),  
borderSide: BorderSide(color: Color(0xffef2e6c) ),

), ),

validator: (value) {  
if (value!.isEmpty) {

return 'Please enter some text'; }

return null; },

onSaved: (val) { email = val;

}, ),

SizedBox( height: 20,

),

// password

TextFormField(

// initialValue: 'Input text',

decoration: InputDecoration( labelText: 'Password', labelStyle: TextStyle(

color: Color(0xffef2e6c), ),

prefixIcon: Icon(Icons.lock\_outline,color:Colors.black), enabledBorder: OutlineInputBorder(

borderRadius: BorderRadius.all( const Radius.circular(100.0), ), ),

focusedBorder: OutlineInputBorder( borderRadius: BorderRadius.all(

const Radius.circular(100.0), ),

borderSide: BorderSide(color: Color(0xffef2e6c) ), ),  
suffixIcon: GestureDetector(

onTap: () { setState(() {

\_obscureText = !\_obscureText; });

},  
child: Icon(

\_obscureText ? Icons.visibility\_off : Icons.visibility, ),

), ),

obscureText: \_obscureText, onSaved: (val) {

password = val; },

validator: (value) {  
if (value!.isEmpty) {

return 'Please enter some text'; }

return null; },

),

SizedBox(height: 30),

SizedBox(  
height: 54,  
width: 184,  
child: ElevatedButton(

onPressed: () {

// Respond to button press

if (\_formKey.currentState!.validate()) { \_formKey.currentState!.save();

AuthenticationHelper()  
.signIn(email: email!, password: password!) .then((result) {

if (result == null) { Navigator.pushReplacement(context,

MaterialPageRoute(builder: (context) => MyApp()));

} else { ScaffoldMessenger.of(context).showSnackBar(SnackBar(

content: Text(  
result,  
style: TextStyle(fontSize: 16),

), ));

} });

} },

style: ElevatedButton.styleFrom( shape: RoundedRectangleBorder(

borderRadius: BorderRadius.all(Radius.circular(24.0))), backgroundColor: Color(0xffef2e6c)),  
child: Text(

'Login',

style: TextStyle(fontSize: 24), ),

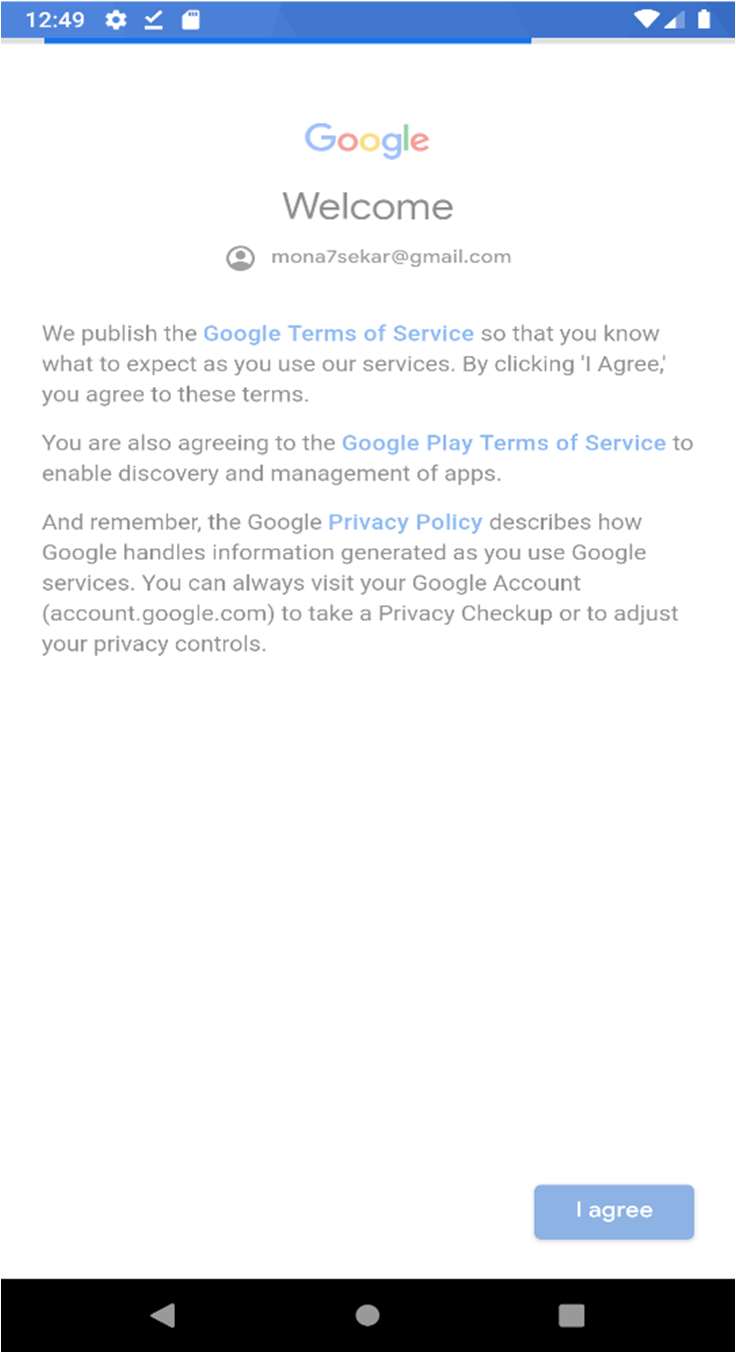
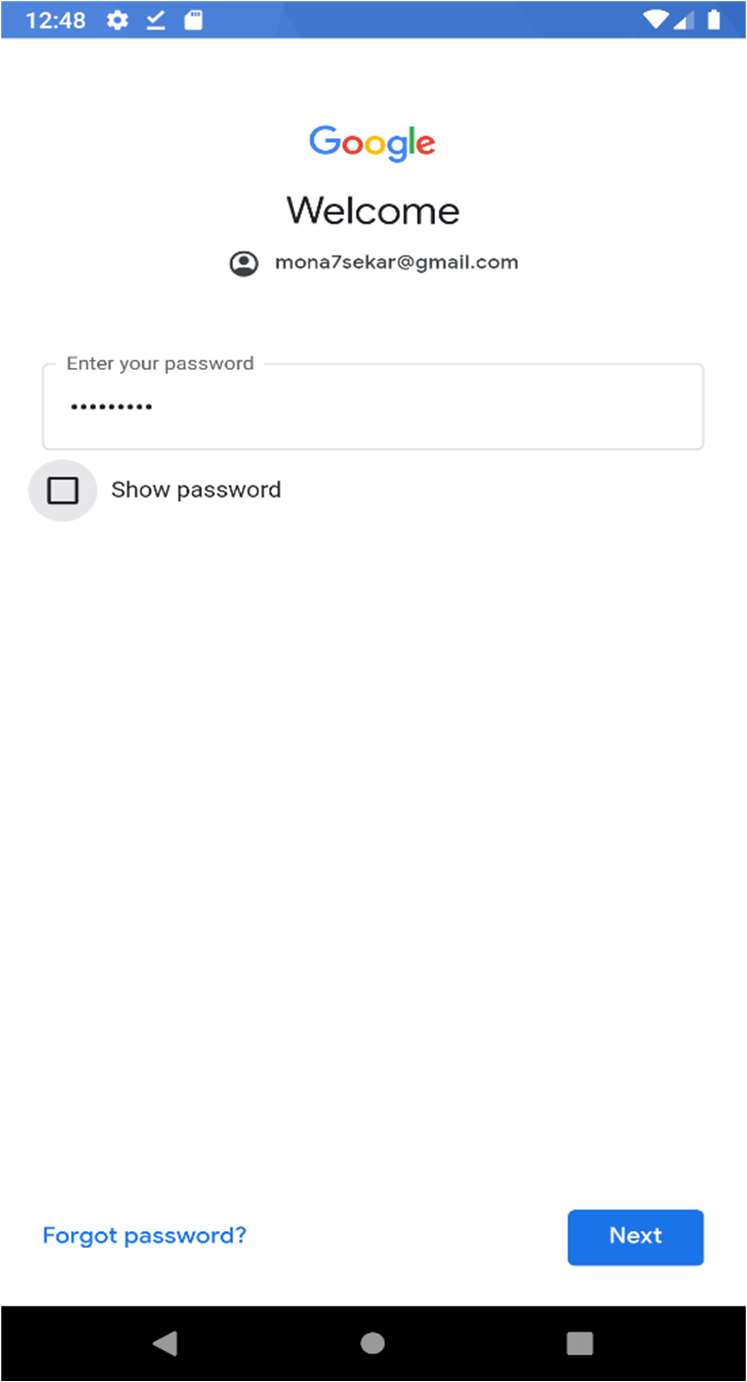
), ),

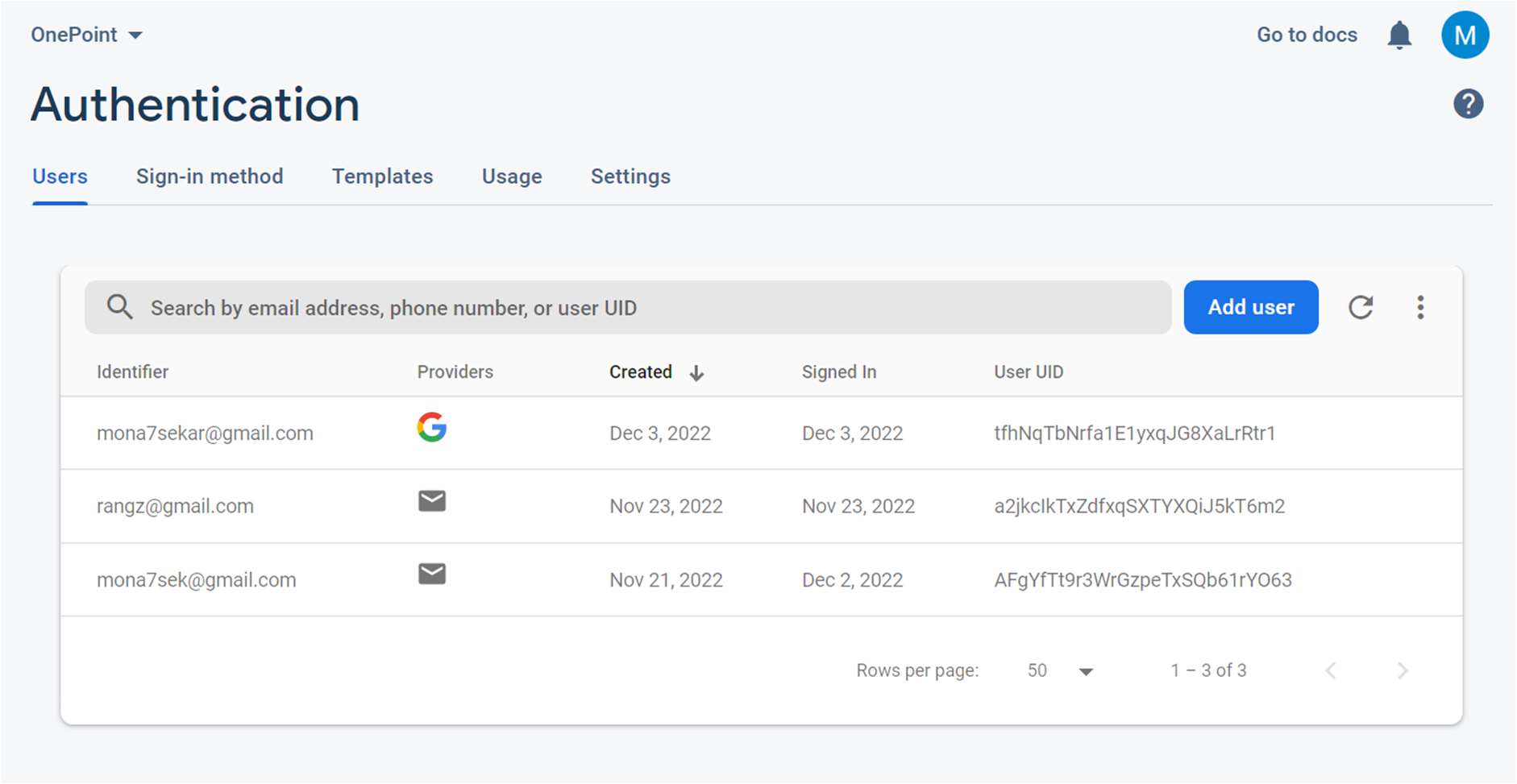
], ),

); }

}

**OUTPUT:**





**RESULT:**

Thus, an application that uses social networking software (Google) for authentication has been implemented.